IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 February 2025, accepted 25 March 2025, date of publication 28 March 2025, date of current version 10 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3555873

RESEARCH ARTICLE

IIoT's Risk Odyssey: Navigating the Risk Propagation of Illegal Information Flows

ARGIRO ANAGNOSTOPOULOU[1], IOANNIS MAVRIDIS[2], MICHAEL ATHANASOPOULOS[1], ALEXIOS MYLONAS[3], AND DIMITRIS GRITZALIS[1]

[1]Department of Informatics, Athens University of Economics and Business, 10434 Athens, Greece
[2]Department of Applied Informatics, University of Macedonia, 54636 Thessaloniki, Greece
[3]Cybersecurity and Computing Systems Research Group, Department of Computer Science, University of Hertfordshire, AL10 9EU Hatfield, U.K.

Corresponding author: Alexios Mylonas (a.mylonas@herts.ac.uk)

ABSTRACT Industrial Internet of Things (IIoT) refers to a broad network of low-cost, interconnected devices, including actuators, programmable logic controllers (PLCs), and sensors. Such environments are characterized by the vast amount of data exchanged among a wide range of devices, applications, and services. The scalability and decentralized nature of IIoT introduces considerable challenges for traditional security mechanisms. As a result, it is crucial to establish more robust security measures, enforce more effective access control policies, and efficiently manage information flows within business processes. In our prior research, we introduced a methodology for the assessment of information flows in IIoT environments and the detection of the illegal ones. Specifically, we utilized a risk-based methodology to model complex business processes as directed graphs. This approach enabled us to thoroughly analyze the interdependencies among participating objects. Through this analysis, we aimed to identify objects that are susceptible to initiating or being influenced by illegal information flows. In our current study, we investigate the propagation of the risk of illegal information flows within and across business processes. Finally, we apply centrality metrics to identify critical objects that require more efficient access control rules and policies in order to mitigate illegal information flows within the IIoT network. To the best of our knowledge, no previous research has explored the concept of risk-based detection of illegal information flows and examined potential propagation of risk in industrial environments.

INDEX TERMS Access control, dependency chain analysis, graph centrality, industry 4.0, information flow control, information security.

I. INTRODUCTION

Internet of Things (IoT) enables ubiquitous internet connectivity, transforming ordinary items into connected, also called smart, objects. These objects are capable of sensing their environment, collecting and processing data, and providing feedback to their surroundings [1]. However, connecting ordinary devices to the Internet exposes them to potential cyber threats. The emergence of Industry 4.0, along with the increased number of internet-connected devices and the growing number of cybersecurity incidents, highlights the importance of enhancing cyber resilience [2].

The associate editor coordinating the review of this manuscript and approving it for publication was Barbara Masucci.

In large-scale Industrial IoT (IIoT) environments is important to secure each interconnected device from unauthorized access. This increases the need to enforce robust access control policies tailored to IIoT objects. Access control systems play a fundamental role in determining which active entities (subjects) are capable of gaining access to passive entities (objects), such as resources. Each subject is granted with a set of permissions regarding a specific object [3], [4], [5]. As these devices interact and exchange data, they either generate or participate in information flows. However, traditional access control models often fall short in governing the transmitted information. This has led to the emergence of information flow control techniques, which are designed to prevent data within an object from being made available

to an unauthorized subject, ensuring that only users with the appropriate rights can access the data [6].

A. CONTRIBUTION

This manuscript builds upon and significantly expands our previously established risk-based methodology for identifying and evaluating illegal information flows within IIoT environments. In our earlier work, we focused mainly on identifying first-order dependencies [7]. However, our current research introduces two significant advancements that deepen the analysis and broaden the capabilities of our approach:

1) **Multi-order dependency analysis.** We extend our earlier methodology to evaluate multi-order dependencies. This enhancement, as outlined in Phase 2, Step 5, allows us to assess cumulative risks more thoroughly by considering indirect relationships between interconnected objects within the IIoT network. By analyzing these dependency chains in greater detail, our method reveals hidden vulnerabilities that first-order analyses often overlook, providing a more comprehensive view of risk propagation throughout the network.
2) **Detection of critical illegal information flow initiators using graph centrality metrics.** We introduce graph centrality metrics to deepen the analysis by quantifying the influence of individual objects within the network. In Phase 3 of the proposed methodology, we apply normalized closeness centrality and Bonacich (eigenvector) centrality to identify critical objects that may not appear vulnerable in standard dependency chain analysis. However, these objects may still exert significant influence due to their central position in the network. This extension enhances our ability to identify key risk objects essential to the stability and security of the infrastructure, offering a more robust and comprehensive framework for risk mitigation.

B. STRUCTURE

The remainder of this document is organized as follows. In Section II, we review related research on methods for identifying illegal information flows. Section III provides the essential background information that underpins our proposed approach. In Section IV, we detail the phases of our methodology, including underlying assumptions made during its development. Finally, in Section V, we assess the effectiveness of our approach and discuss the findings in detail. Section VI provides a detailed analysis of how our approach differentiates itself from related work in the field. The conclusions, limitations and future work are exhibited in Section VII. Finally, Appendix A, B, and C illustrate the detailed calculations of the formulae included in our approach.

II. RELATED WORK

In recent years, there has been an increased interest in the concept of information flow control, also known as information flow security. Information flow control is related with the propagation of information in a program during execution [8]. However, there is a prevalent misconception that information flow control acts exactly as access control. This is invalid, since access control determines whether a subject can manipulate an object based on his/her granted access rights.

The research community tries to develop novel methodologies that can handle information flows in order to enhance the confidentiality and integrity of the propagated information. Information flow control can be implemented at various levels, including language level, operating system level, and hardware level. In this section, we present the existing published work of several research groups that tried to identify and control information flows.

Numerous studies emphasize on the implementation of information flow control on hardware level through information flow tracking. This technique focuses on security vulnerabilities that are based on factors such as flaws on the design, verification, testing, manufacturing, and deployment of the hardware [9], [10], [11]. For example, Zhang et al. developed a hardware design language in order to analyze information flows at the hardware level in a static way. Their aim was to prove that it is feasible to design complex hardware equipment with information flow security in mind [12].

Some research groups follow a programming language-centric approach, e.g. the development of policies that manage how the information should flow [13], [14], [15], [16], [17]. The conventional security mechanisms, such as access control, cannot adequately handle information flows. Thus, a good practice is the use of programming-language techniques to establish and enforce information flow policies [14].

However, the research community has made significant efforts to develop risk-based access control methodologies that address deficiencies in policy enforcement and improve detection of unauthorized information flows. In IIoT environments, where there is a great number of data exchanges, risk-based models can effectively manage illegal information flows through the analysis of data transfer between components. Such models assign risk scores to information flows and access requests, to enhance detection and mitigation of illegal flows, ensuring confidentiality and integrity. This flexible, context-sensitive approach allows access permissions to adapt based on real-time risk factors such as resource sensitivity, device security state, and potential threats.

Several studies contribute to this field. Khambhammettu et al. [18] propose a risk-based framework that quantifies risk as a product of threat and impact scores, proposing four methodologies for threat assessment based on object sensitivity and subject trustworthiness. It aligns with the NIST SP 800-30 standard for risk management and incorporates situational factors, emphasizing the importance of contextual judgment in selecting threat assessment

methods. Ni et al. [19] explore the application of fuzzy inference in risk-based access control, addressing limitations in traditional binary controls. The propose scalable solutions, including criteria for choosing fuzzy operations, and a token-based system to limit potential damage for critical environments. Zhang et al. [20] present the Benefit and Risk Access Control (BARAC) model, which emphasizes on balancing the risks of information disclosure with the benefits of information sharing. BARAC dynamically adjusts access based on operational needs, especially in high-risk scenarios such as battlefield operations.

Shaikh et al. [21] propose two dynamic decision-making methods using trust and risk values derived from user behavior, to adapt access decisions in real-time. The first method integrates a simple risk-based decision approach, while the second employs an Exponentially Weighted Moving Average (EWMA) to prioritize recent behavior, allowing for a more responsive system. Atlam and Wills [22] present a novel risk estimation technique which integrates fuzzy logic with expert judgment to dynamically assess security risks related to access requests. The validation included expert interviews and simulations, highlighting its precision in diverse applications. In another study, Wills and Atlam [23] develop an Adaptive Neuro-Fuzzy Inference System (ANFIS) to enhance risk estimation in smart homes, combining neural networks and fuzzy logic. Their approach dynamically estimates security risks based on user context, resource sensitivity, action severity, and risk history, allowing for real-time adaptation to changing conditions. Aliyu et al. [24] proposed an Adaptive Risk-based Access Control System (ad-RACs) for edge computing environments. They evaluate access requests using four factors: user context, resource sensitivity, action severity, and risk history. The system applies CatBoost to estimate risk and enforces access rules using the Chinese Wall policy. It supports real-time decision-making, adapting to user behavior over time. Authors assess ad-RACs through simulations, achieving higher recall and F1 score than existing methods.

Finally, Chen and Crampton [25] extend Role-Based Access Control (RBAC) with risk-aware elements, incorporating user trustworthiness and role competence for more fine-grained and context-aware access control in sensitive environments. Authors propose three simple models for integrating risk into RBAC, along with a comprehensive risk-aware RBAC model that combines features of the simpler models in an attempt to provide a balance between access flexibility and security.

Our approach assesses the information flows that occur in a smart grid based on business processes modelling and detects the illegal ones. Our research adopts a similar framework to [26] and [27], but applies different formulae in a totally different context. The detection of the illegal flows is based on a risk-based approach we presented on our previous work [7]. In particular, we represent business processes as a graph and evaluate the risk associated with each information flow to detect the objects that are likely to participate in an illegal information flow. We examine how the risk of an illegal information flow in one transaction is propagated to subsequent transactions within a business process. Also, we use centrality metrics to identify objects that may not be obviously prone to participate to illegal information flows through standard dependency chain analysis. To the best of our knowledge, our work represents the first attempt in the literature that integrates the concepts of access control, information flow control, risk propagation, and centrality metrics, providing a comprehensive risk-based methodology for the detection of illegal information flows in IIoT environments.

III. THEORETICAL BACKGROUND

In this section we explain the fundamental concepts that our approach is based on. We specifically discuss the definitions of access control and information flow control, as well as their importance for securing digital ecosystems from illegal information flows and unauthorized access. These concepts are essential for ensuring the confidentiality and integrity of information, particularly within the context of IIoT and critical infrastructure environments.

A. ACCESS CONTROL

Access control remains a key element of system security, since it defines the rules through which users or processes interact with system resources. The main components of an access control system are: (i) policies, which specify the conditions under which a user or a process can access a resource, (ii) models, which specify how an organization can allocate permissions to subjects and enforce access policies, and (iii) mechanisms, which are responsible to grant or deny access to subjects based on predefined criteria [5], [7].

Organizations can choose from a wide range of available access control models, each designed to fulfill specific security requirements. Among the most prevalent models are:

- *Lattice-Based Access Control (LBAC)*: Assigns a security class for each object. As a result, when an information flow occurs between two objects, let suppose between object a and object b, we record that there is an information flow from the security class of object a to the security class of object b [28], [29].
- *Mandatory Access Control (MAC)*: Follows two main principles: (a) No-read-up, where a low-level subject is not allowed to read information from high-level objects, and (ii) No-write-down, where high-level objects can only be written by low-level objects. These principles aim to ensure that the information flows only upward, from the lower levels to the higher ones [30].
- *Discretionary Access Control (DAC)*: Defines that each object has a subject as an owner. The owner is responsible to define access rights for its object and is capable of transferring access rights to other subjects [31].

- *Role-Based Access Control (RBAC)*: Assigns roles according to the responsibilities and the job description of users. The roles can dynamically change as new systems and applications are added to the infrastructure, a user may get promoted and the responsibilities may be different, or a user may be degraded, and some permissions should be revoked from this role [5], [32].
- *Capability-Based Access Control (CapBAC)*: Assigns capability tokens to subjects. A subject can issue a capability token as the owner of a device (object). A token refers to a specific subject and includes the access rights that the subject is granted [3].

Generally, access control works as follows: let suppose that we have an object O that manipulates a subject S. The following tuple describes an access control rule: <S,O,OP>, where OP denotes an operation (e.g., read, write) that a subject can perform, and the pair <O,OP> denotes the access right that the subject is granted in regard to object O [33].

B. INFORMATION FLOW CONTROL

An information system is composed of two types of entities: (i) objects, encapsulation of data and operations to manipulate data, and (ii) subjects such as users that issue operations on an object to manipulate the data [33]. Security engineers should protect the confidentiality and integrity of the propagated information. This can be achieved by setting appropriate rules that define the acceptable paths for the flow of information between entities [8], [14]. Enforcing proper access control rules enhances information security, by eliminating unauthorized access to objects and resources.

Let assume that we have subjects S_i and S_j. Subject S_i is granted with two access rights: (1) <f, read> which permits S_i to read data from object f, and (2) <g, write> that permits S_i to write data to object g. Subject S_j is granted only with the access right <g, read> which permits S_j to read data from object g. The actions of "read" and "write" are operations that an object can perform to a subject. Thus, a subject is granted with an access right that defines in which object the subject can perform an operation. All these drive to legal information flows since they follow the access control rules. Now let assume that the subject S_j wants to read data that is stored in object f. This is not feasible due to the enforced access control rules. However, S_j can get access to these data by just reading the data stored in object g (where subject S_i writes these data). Thus, information of object f illegally flows into S_j through acceptable actions performed by S_i [33].

A transaction $T_{(x\to y)_i}$ is defined as a distinct operation occurred between two objects. Each transaction can be characterised as *Good* or *Bad*. Tracking all transactions between two specific objects can help determine whether an information flow $IF_{x\to y}$ between these objects has positive or malicious intent. Each information flow can link to numerous associated transactions. In other words, an information flow is defined as a relationship between two objects. In this work, we examine two groups of transactions: legal and illegal ones. In the first group we have two categories: (i) *legal read*, and (ii) *legal write*, whose meaning is quite straightforward. The second group includes four categories [33]:

1) *Illegal read*: a subject reads data from an object, without being granted with the corresponding permission.
2) *Illegal write*: a subject writes data to an object after illegally reading data from another object, without being granted with the corresponding permission.
3) *Suspicious read*: a subject reads data from an object whose data is not allowed to be brought to other objects.
4) *Impossible write*: a subject writes data to an object after suspiciously reading data from another object, without being granted with the corresponding permission.

Fig. 1 illustrates an indicative example of how the illegal transactions work [7], [33].

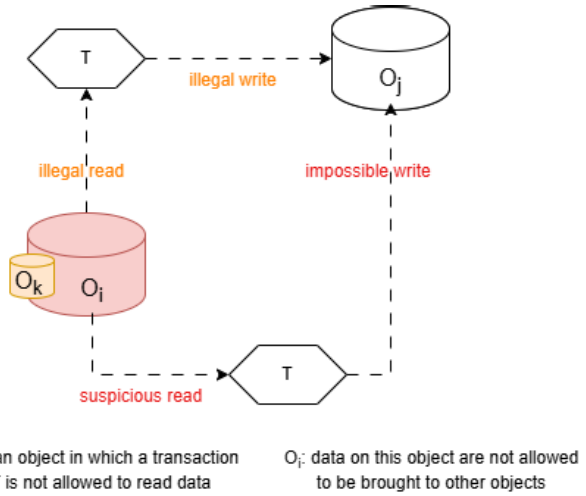


FIGURE 1. Definition of the illegal types of transactions [33].

IV. RISK-BASED INFORMATION FLOW CONTROL METHODOLOGY

The proposed methodology builds upon a previously developed, by our research team, risk-based approach to model interdependencies in critical infrastructures [26], [27]. Specifically, in the current modified and expanded methodology, we assess each information flow and identify the illegal ones within IIoT environments. Our objective is firstly to identify the objects that are most likely to participate in illegal information flows, considering both the direct and indirect flows between objects and afterward to examine how this risk is propagated within a business process. Finally, for all the objects we take into consideration aspects related to graph centrality metrics. Incorporating metrics like eigenvector centrality and closeness centrality is crucial for a comprehensive risk analysis, since they reveal objects that have a great impact on information propagation. This offers important insights into the structure and operation of the modelled network. Through the examination of these objects, we can identify potential points of vulnerability that can be used to facilitate illegal information flows.

This approach ensures that security measures are tailored to protect the infrastructure's most important objects, enhancing its resilience.

The proposed methodology consists of three distinct phases, each containing particular steps as described below.

Phase 1 - Attack graph modelling: This phase involves the modelling of business processes as a directed graph. These graphs are further annotated to depict potential attack vectors, indicated by a red flag called ''Bad''.

Phase 2 - Graph risk analysis: This phase focuses on calculating the cumulative risk of all detected (direct or indirect) information flows so as to identify the dependency chains. This comprehensive risk calculation facilitates the ranking and prioritization of attack paths based on their respective risk levels and potential to impact data integrity.

Phase 3 - Centrality Metrics Calculation: This phase focuses on aspects related to graph centrality metrics. Specifically, we calculate Bonacich (Eigenvector) and Closeness centrality for each node (object in our context) of the directed graph. Centrality metrics quantify the relative importance of nodes by evaluating their position and connections within the network. Objects with higher centrality values have a greater influence on the overall network, making them key candidates for risk mitigation measures. In this phase, we assess the potential involvement of an object in illegal information flows, even in cases where standard dependency chain analysis may fail to identify it as a potential threat.

In this subsection we provide a thorough description of the fundamental assumptions that led us to develop, implement, and evaluate the proposed methodology. The following assumptions help us build our methodology and ensure that it is consistent with realistic limitations and requirements of IIoT. The assumptions are classified into two distinct groups: (a) the preliminary, and (b) the advanced ones.

1) Preliminary Assumptions

This category includes the assumptions that, while critical to our methodology, are relatively straightforward and are based on established precedents or logical deductions:

- *Value Scaling*. Given the absence of a directly comparable risk-based approach to the field of information flow control, we have rescaled the numerical values of our methodology based on our knowledge and prior experience as risk assessment professionals.
- *Selection of Centrality Metrics*. To enhance our methodology, we applied graph centrality metrics to objects. According to literature [34], we decided that the Normalized Closeness Centrality, and the Bonacich (Eigenvector) Centrality are the most suitable centrality metrics according to the objectives of our approach.

2) Advanced Assumptions

This category includes more complicated assumptions which are essential to the evaluation of our methodology:

- *Dataset Construction*. Due to the fact that our approach refers to critical infrastructures, and especially smart grids, obtaining real-world datasets is challenging since they are inaccessible due to their confidential nature [35]. User-defined datasets can be employed for training and testing purposes; however, they typically do not encompass all possible attributes of real network data and may exhibit biases in data selection. Public datasets such as BATADAL [36], SWaT [37], and WADI [38] primarily concentrate on SCADA systems within power grids and fail to provide specific features related to power system measurements, protocols, or new smart devices like smart inverters, IEDs, and communication infrastructure. Therefore, to address this limitation, we decided to study real-world use cases, model them as dependency graphs, and create a sufficient volume of data in order to feed them on the CIDA tool [39]. We modified and expanded the functionality of the tool to model and analyze information flows among objects [40]. An earlier version of the presented graphs has been published in our previous work [41].

A. PHASE 1 - ATTACK GRAPH MODELLING

A dependency graph can be used in order to represent the business processes of a critical infrastructure. These graphs help at depicting the information flows and transactions between the objects of the IIoT network. The graphs are directed and weighted, signifying the direction and significance of the information passing from one object to another. In these graphs, the nodes symbolize the network's objects, while the edges indicate the information flows. To construct these graphs, we employ the following representations: (i) **O** for the set of objects (nodes) within the IIoT network, (ii) **IF** for the set of information flows between objects, and (iii) **T** for the set of transactions corresponding to each information flow.

In our approach, we assumed that an object O encapsulates data and supports a basic operation OP (e.g., read or write). An edge $O_x \rightarrow O_y$ depicts an information flow $IF_{x\rightarrow y}$ from object O_x to object O_y. A transaction is an individual instance of information being transferred from one object to another. Each transaction can be seen as a discrete step within the broader information flow between two objects. Multiple transactions $T_{(x\rightarrow y)_i}$ can occur within a single information flow $IF_{x\rightarrow y}$. This means that a single information flow is not limited to a single transaction. Instead, it can encompass multiple transactions, indicating a complex and dynamic exchange of data between two objects.

Each transaction depicts a ''read'' or a ''write'' operation to an object. A graph includes multiple details, such as the category of the data transmitted (e.g. sensor data, configuration data etc.), and the type of operation (e.g. read, write) for each transaction. In Fig. 2 we present an example of a dependency graph that is composed of the following objects: O_x, O_y, O_u, and O_z. Here, O_x writes *sensor data* to O_y and reads *configuration data* from O_u. Meanwhile, O_z writes *configuration data* to O_u and reads *customer data* from O_u.

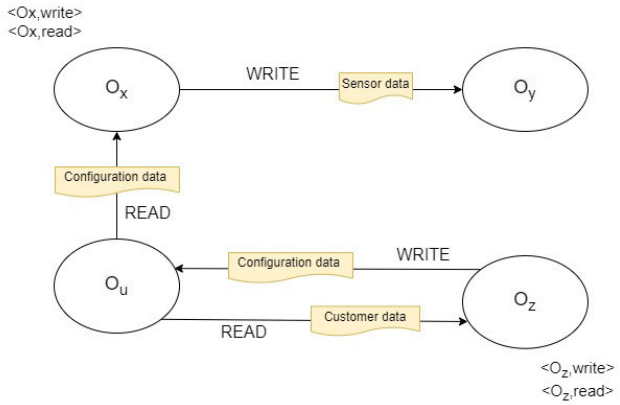


FIGURE 2. Example of a dependency graph [7].

B. PHASE 2 - GRAPH RISK ANALYSIS

The second phase of our approach is consisted of five steps:

1) **Supporting Metrics Calculation**: We calculate the metrics of (a) severity, (b) operation factor, and (c) legality for each information flow.
2) **Likelihood Calculation of Illegal Information Flows**: We estimate the probability of illegal flows occurrences based on existing graph connections.
3) **Transaction Impact Calculation**: We assess the impact of an illegal transaction.
4) **Risk Calculation of Illegal Information Flows**: We calculate the overall risk of illegal information flows.
5) **Cumulative Dependency Risk Calculation**: We compute cumulative risk of dependency paths in the graph.

Steps 1 to 4 calculate the first-order dependencies in a business process. A first-order dependency exists when there is a direct information flow or interaction from one object to another. For example, if object A directly reads data from object B, the dependency from A to B is a first-order dependency. Identifying first-order dependencies is crucial because these direct relationships can be the initial points of risk propagation. Understanding these can help in implementing direct security measures between these objects. Step 5 computes the n-order dependencies. An n-order dependency exists when the relationship between two objects involves intermediate objects. For example, if object A reads data from object B, which then writes data to object C, the dependency of object C on object A (through object B) is a second-order dependency. This concept can be extended to higher orders (third-order, fourth-order, etc.), where more intermediate objects are involved. Understanding n-order dependencies is essential for comprehensive risk analysis because risks can propagate through multiple objects and pathways. For instance, a vulnerability in object A could indirectly affect object C if there are intermediate objects (object B) facilitating the risk transfer. By analyzing these multi-order dependencies, we can uncover hidden vulnerabilities that are not apparent when looking only at direct relationships.

TABLE 1. Values assigned to the severity metric [7].

Category of data	Severity
Acknowledgement data	1
Meteorological data	1
Customer data	2
Billing & Pricing data	3
Sensor data	4
Configuration data	5

TABLE 2. Values assigned to the operation factor [7].

Type of operation	Operation factor
read	2
write	3

TABLE 3. Values assigned to the legality metric [7].

Category of transaction	Legality
Legal Read	1
Legal Write	1
Illegal Read	2
Illegal Write	3
Suspicious Read	4
Impossible Write	5

1) STEP 1: SUPPORTING METRICS CALCULATION

In this step we calculate three supporting metrics for each information flow depicted in the graph.

a: SEVERITY

It is important for each transaction to consider the category of information that flows from one object to another. Depending on this category we have defined a severity value, presented on Table 1. The provided list can be adapted in accordance with the particular infrastructure and context in which the suggested methodology will be used. The values of Severity range from 1 to 5, with 1 representing a very low severity and 5 indicating a very high severity.

b: OPERATION FACTOR

For each transaction, we consider that there are two possible operations that an object can perform, either "read" or "write". For each type of operation, we have assigned a distinct value to the operation factor, based on the potential impact each operation could have on the infrastructure if executed by a malicious actor. Table 2 presents the values of this metric based on the operation type of a transaction.

c: LEGALITY

We classified the transactions into six categories based on their legality: (1) Legal Read, (2) Legal Write, (3) Illegal Read, (4) Illegal Write, (5) Suspicious Read, and (6) Impossible Write. These categories are explained in Section III-B. As presented in Table 3, the values of the Legality metric are ranging from 1 to 5, with 1 representing a totally legal transaction and 5 indicating a totally illegal one.

2) STEP 2: LIKELIHOOD CALCULATION OF ILLEGAL INFORMATION FLOWS

For every relationship between two objects, we calculate a likelihood value, which assess the probability of an illegal

information flow to occur. Since this metric is a probability, the range of its values is [0,1] indicating a prediction on whether an object that initiates an information flow has a malicious intent. Specifically, we symbolize with $IF_{x\to y}$ an information flow that occurs between the objects O_x and O_y. For such an information flow, every transaction $T_{(x\to y)_i}$ in notated as "Good" if the flow is legal, and as "Bad" in case that the flow under examination exhibits an illegal behavior. Formula 1 estimates the *Illegal Information Flow Likelihood* (denoted as $IIFL_{x\to y}$).

$$IIFL_{x\to y} = \frac{\sum_{i=1}^{n}(T_{(x\to y)_i} \textit{ marked as "Bad"})}{\sum_{i=1}^{n}(T_{(x\to y)_i})} \quad (1)$$

where:

- n represents the number of transactions that occurred between two objects.
- $\sum_{i=1}^{n}(T_{(x\to y)_i}$ *marked as* "*Bad*"$)$ counts the number of illegal transactions (those marked as "Bad") between O_x and O_y.
- $\sum_{i=1}^{n}(T_{(x\to y)_i})$ represents the total number of transactions between O_x and O_y.

3) STEP 3: TRANSACTION IMPACT CALCULATION

For each transaction $T_{(x\to y)_i}$ we calculate the impact (denoted as $TI_{(x\to y)_i}$) it has on the business process. For instance, a high value for this metric indicates that a security incident would significantly impact the operation of the relevant business process and, consequently, the entire critical infrastructure.

Since there is no existing standard for computing such values, we developed Formula 2 to determine the impact of each transaction.

$$TI_{(x\to y)_i} = Severity \times Operation\ factor \times Legality \quad (2)$$

By considering all possible combinations of the parameter values in Formula 2, we concluded that the range of the Transaction Impact (TI) is [2,75]. In order to easier understand and compare the results of our methodology, we decided to rescale these values into the range [0,10] using Formula 3.

$$f(I) = \frac{(b-a)(I-min)}{max-min} + a \quad (3)$$

where:

- I represents the original transaction impact value that needs to be scaled. In the context, this is the calculated impact of a specific transaction.
- min is the minimum value in the range of the original transaction impact values. It represents the lowest impact value observed in the dataset.
- max is the maximum value in the range of the original transaction impact values.
- a is the lower bound of the desired scaled range.
- b is the upper bound of the desired scaled range.

For instance, let assume that the original transaction impact values range from 2 to 75 (min = 2, max = 75), and we want to scale this to the range [0, 10] (a=0, b=10). For an original impact value I, we would use Formula 3 as follows:

$$f(I) = \frac{(10-0)(I-2)}{75-2} + 0 = \frac{10*(I-2)}{73} \quad (3a)$$

TABLE 4. Rescaling of the transaction impact values [7].

Calculated TI Values	STI Values	STI Level
[2,10)	[1,2)	Very Low
[10,26)	[2,4)	Low
[26,42)	[4,6)	Medium
[42,58)	[6,8)	High
[58,75]	[8,10]	Very High

TABLE 5. Levels of illegal information flow risk [7].

Illegal Information Flow Risk value	Illegal Information Flow Risk level
[0,2)	Very Low
[2,4)	Low
[4,6)	Medium
[6,8)	High
[8,10]	Very High

In Table 4 we present the Scaled Transaction Impact values (denoted as STI). We further classify these values into five distinct levels. Finally, based on the calculated impact for the individual transactions between two objects 0_x and O_y, we estimate the Total Transaction Impact (denoted as $TTI_{x\to y}$) of the information flow $IF_{x\to y}$ in Formula 4 as the average transaction impact of all the transactions between objects 0_x and O_y.

$$TTI_{x\to y} = \frac{\sum_{i=1}^{n} TI_{(x\to y)_i}}{n} \quad (4)$$

4) STEP 4: RISK CALCULATION OF ILLEGAL INFORMATION FLOWS

By combining the values of $IIFL_{x\to y}$ and $TTI_{x\to y}$ we can estimate the risk of an information flow $IF_{x\to y}$. Risk highlights information flows with elevated risk levels and consequently the objects that are more prone to be involved in an illegal information flow. Specifically, security experts should focus on enhancing the controls (such as access control rules, etc.) already applied to these objects in order to ensure that they will be more precise and stringent. The risk of an illegal information flow, indicated as $R_{x\to y}$, for each information flow $IF_{x\to y}$ is computed using Formula 5.

$$R_{x\to y} = TTI_{x\to y} \times IIFL_{x\to y} \quad (5)$$

In Table 5 we classify the ranges of possible values of $R_{x\to y}$ to levels to easily determine whether a risk is acceptable or not. To address this question, we need to evaluate the extent to which the infrastructure can tolerate the risk, considering the criticality of its systems and business processes. However, the fundamental principle is that the lower the risk is, the better it is for the security of the infrastructure.

TABLE 6. Levels of cumulative illegal information flow risk [7].

Cumulative Risk value	Cumulative Risk level
[0,1)	Very Low
[1,5)	Low
[5,10)	Medium
[10,15)	High
≥ 15	Very High

5) STEP 5: CUMULATIVE DEPENDENCY RISK CALCULATION

While first-order dependencies highlight the direct risks between objects, n-order dependencies provide a broader perspective, uncovering potential indirect risks that could propagate through the infrastructure. It is crucial to understand and analyze both types of dependencies in order to develop robust security frameworks for IIoT. A business process may include a great number of information flows in order to accomplish a task. Our goal is to examine how the risk of an individual information flow can impact the rest flows and whether it magnifies the overall illegal information flow risk of the business process. Formula 6 calculates the dependency risk of illegal information flows in a series of information flows.

$$R_{IF_0,\ldots,IF_n} = L_{IF_0,\ldots,IF_n} \times I_{IF_{n-1},IF_n} = (\prod_{i=1}^{n} L_{IF_{i-1},IF_i}) \times I_{IF_{n-1},IF_n} \quad (6)$$

Specifically, let assume that $IF_0 \rightarrow IF_1 \rightarrow IF_2 \rightarrow \ldots \rightarrow IF_n$ is a chain of n information flows, based on specific business processes in a smart grid. Also, we suppose that $IIFL_{IF_0,\ldots,IF_n}$ is the likelihood of the n^{th}order cumulative dependency risk for the chain of information flows from IF_0 to IF_n. The I_{IF_{n-1},IF_n} refers to the impact of the illegal information flow between the last two objects in the dependency chain, $I_{IF_{n-1}}$ and IF_n (e.g., the $IF_{n-1} \rightarrow IF_n$ dependency).

Formula 7 calculates the cumulative dependency risk of the chain $R_{IF_0,\ldots,IF_n}$ due to the n^{th}-order dependency. The cumulative dependency risk (denoted as $DR_{IF_0,\ldots,IF_n}$) considers the overall dependency risk exhibited by all the business processes through the entire chain of the n^{th}-order dependencies.

$$DR_{IF_0,\ldots,IF_n} = \sum_{i=1}^{n} R_{IF_0,\ldots,IF_n} = \sum_{i=1}^{n}((\prod_{j=1}^{i} L_{IF_{j-1},IF_j}) \times I_{IF_{i-1},IF_i}) \quad (7)$$

The cumulative dependency risk is calculated as the sum of the dependency risks of the affected objects in the chain, due to illegal information flows occurred in a business process of the dependency chain. Table 6 classifies the ranges of the final cumulative illegal information flow risk to levels.

C. PHASE 3 - GRAPH CENTRALITY METRICS

In this section, we explore the centrality metrics applied to our graphs. These metrics help identify objects that may be susceptible to participating in illegal information flows, yet may not be recognized as potential threats through standard dependency chain analysis. Centrality metrics are valuable for detecting objects that have greater influence within the graph [42]. We calculated two centrality metrics: (i) normalized closeness and (ii) Bonacich (eigenvector) centrality.

Centrality metrics are extensively utilized within network connectivity and flow management [43]. In graph theory and network analysis, centrality metrics quantify a node's position relative to others and estimate its relative importance within a graph. In our study, directed risk relations between connected objects are represented by graph edges. Objects with high centrality values significantly impact the overall network. Consequently, such objects are prime candidates for implementing risk mitigation controls.

1) NORMALIZED CLOSENESS CENTRALITY

Closeness centrality measures an object's relative position in a two-dimensional space using geodesic distances [42]. Closeness centrality (C_c) of an object is defined as [44]:

$$C_c(u) = \frac{1}{\sum_{v=1}^{N} d(u,v)} \quad (8)$$

where:

- $C_c(u)$ is the closeness centrality of object u.
- u represents the specific object for which the closeness centrality is being calculated.
- N is the total number of objects in the graph.
- d(u,v) is the shortest path distance between u and v.
- $\sum_{v=1}^{N} d(u,v)$ calculates the total distance from object u to all other objects in the graph.

In our proposed method, we used the normalized closeness centrality, which enables comparison of centrality values across networks of different sizes. Without normalization, the closeness centrality values would be heavily influenced by the size of the network, making it difficult to compare the importance of nodes in networks of different scales [45]. Formula 9 defines the normalized closeness centrality [46].

$$C(u) = \frac{N-1}{\sum_{v=1}^{N} d(u,v)} \quad (9)$$

where:

- $C_(u)$ is the normalized closeness centrality of node (object, in our context) u.
- u represents the specific object for which the closeness centrality is being calculated.
- N is the total number of objects in the graph.
- $N-1$ accounts that object u is excluded from in the shortest path calculations to all other objects.
- d(u,v) is the shortest path distance between u and v.
- $\sum_{v=1}^{N} d(u,v)$ calculates the total distance from object u to all other objects in the graph.

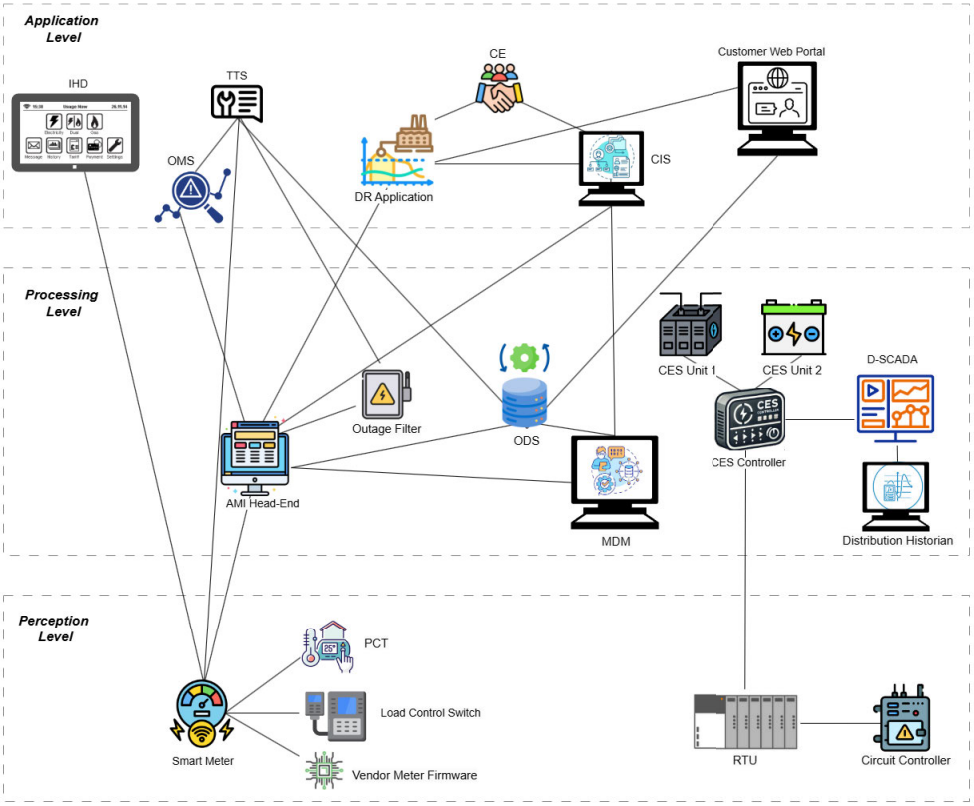


FIGURE 3. System architecture diagram illustrating components for the selected use cases.

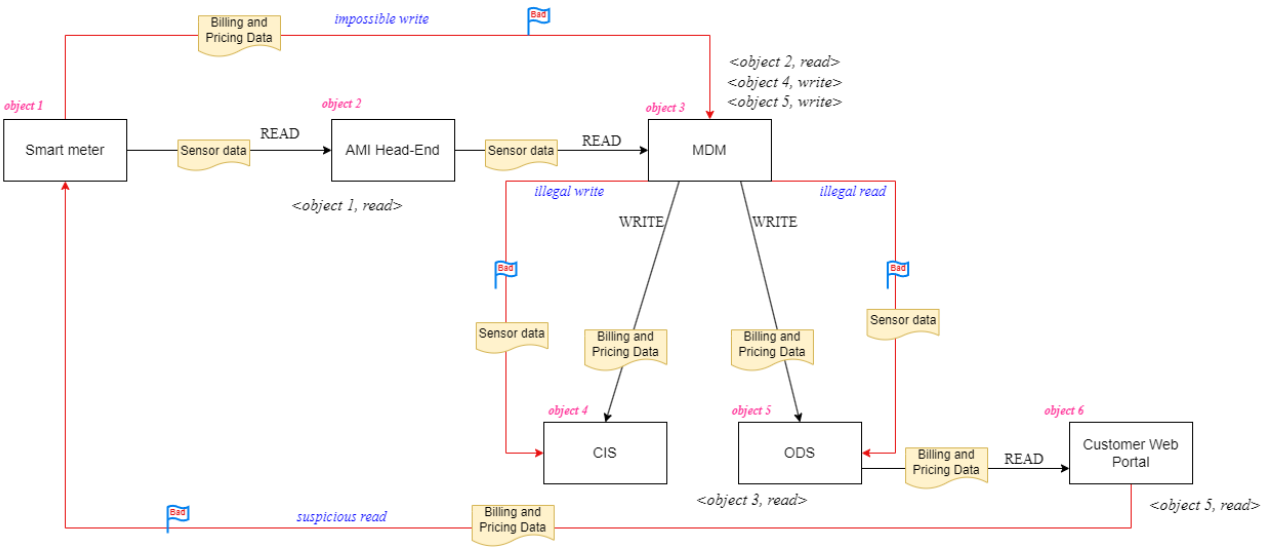


FIGURE 4. Graph representation of the business process: Bulk meter readings.

2) BONACICH (EIGENVECTOR) CENTRALITY

Bonacich (eigenvector) centrality measures the influence of an object in a network. This metric is quite interesting since it indicates the objects that not only cause cascading failures but also cause multiple dependency chains of high risk. When the value of this metric is low, the object does not have many dependencies and does not influence them [34].

$$C_i = \frac{1}{\sqrt{\sum_{\Sigma_j} C_j^2}} x \sum_j A_{ij} x C_j \tag{10}$$

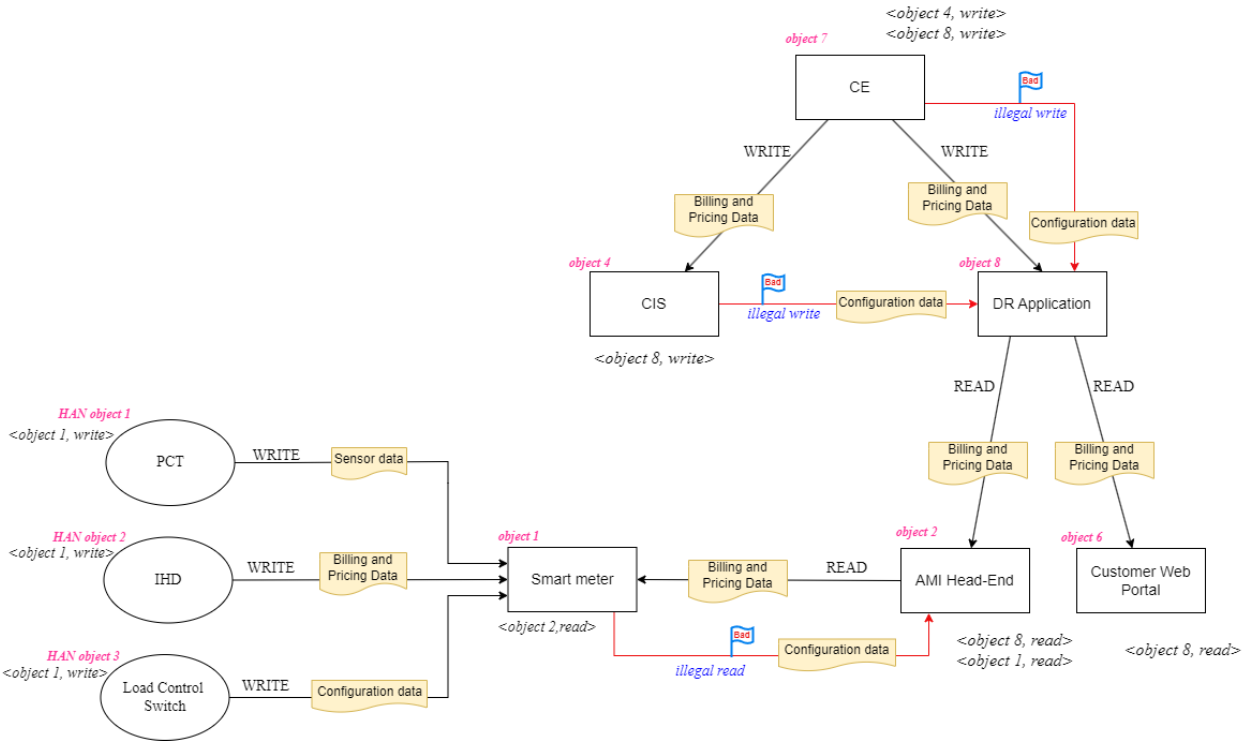


FIGURE 5. Graph representation of the business process: DR HAN pricing and event customer Opt-Out.

TABLE 7. Abbreviations used in use cases.

Abbreviation	Description
AMI	Advanced Metering Infrastructure, used for collecting and transmitting energy usage data.
AMI Head-End	A back-office system responsible for managing and controlling the AMI.
CIS	Customer Information System, a platform that handles customer data and billing operations.
CE	Customer Engagement, which facilitates tailored communications with customers, typically via a Programmable Communicating Thermostat (PCT).
Customer Web Portal	An interactive website that allows customers to view and exchange relevant energy information.
DR Application	A demand response management system that controls load, pricing, and sends messages to devices.
IHD	In-Home Display, a portable screen device providing customers with data such as energy usage, price details, or demand response alerts.
Load Control Switch	An electric switch that can be remotely operated to open or close circuits.
MDM	Meter Data Management, software used for processing smart meter data, including aggregation, validation, estimation, and editing.
ODS	Operational Data Store, a data warehouse designed to store and manage operational data, including metering events and messages.
PCT	Programmable Communicating Thermostat.
RTU	Remote Terminal Unit.
Smart Meter	A digital device that measures electrical parameters like active power, reactive power, and apparent energy for energy management purposes.
Vendor Meter Firmware	A development tool used for programming meters and performing field updates directly on meters.

where:

- C_i is the eigenvector centrality of object i.
- $\sqrt{\sum_{\Sigma_j} C_j^2}$ is the sum of the squares of the centrality scores of all objects in the infrastructure.
- A_{ij} represents the adjacency matrix entry indicating the connection between object i and object j.
- C_j represents the eigenvector centrality of object j. The formula calculates the centrality of object i based on the centrality scores of its neighbors, j.
- The sum is over all objects j connected to object i.
- $\frac{1}{\sqrt{\sum_{\Sigma_j} C_j^2}}$ is a normalization factor that normalizes the centrality scores to ensure that the sum of the squares of the centrality scores equals one.

V. EVALUATION OF OUR APPROACH

In this section we model eight business processes of a smart grid as graphs. Then, we apply the formulae discussed in Section IV-B to these graphs, calculating the first-order and n-order dependencies. Finally, we calculate the metrics of (a) Normalized Closeness Centrality, and (b) Bonacich (Eigenvector) Centrality (as discussed in Section IV-C).

A. ANALYSIS OF USE CASES

We present eight trustful use cases that are referred to real business processes in smart grid environments [40]. For each use case, we give a brief description, and then we depict this information in a graph. Graphs present how information flows in an infrastructure regarding the executed operation. They include details, such as the category of the data transmitted and the type of operation. An earlier version of these graphs appeared in our previous publication [41], however we have enriched them by adding potential illegal information flows. The illegal transactions are annotated with the flag "Bad",

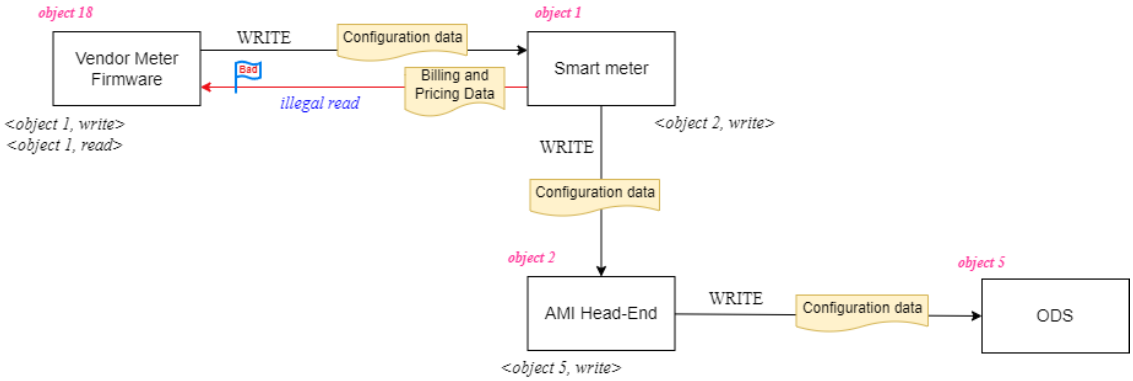


FIGURE 6. Graph representation of the business process: In-Field programming of smart meter and meter firmware upgrade.

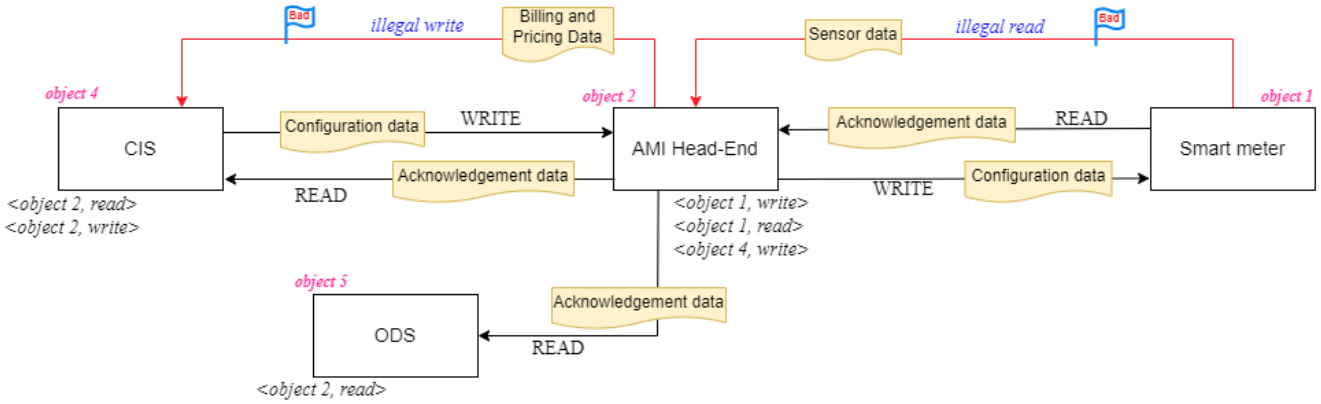


FIGURE 7. Graph representation of the business process: Meter remote Connect / Disconnect.

and the respective operation (e.g., illegal write, illegal read etc.). Finally, Table 7 introduces the abbreviations, along with a brief description of the components that are used in the described scenarios.

Fig. 3 illustrates the overall system architecture encompassing the components involved in the selected use cases. The depicted architecture does not represent a comprehensive smart grid environment but rather focuses on the elements relevant to the use cases. There are several architectures for the IoT and each one provides a different level of abstraction [47]. Each layer is classified by its purpose and defined according to the devices that operate within. The presented architecture is organized into three levels: Perception Level, Processing Level, and Application Level, reflecting the flow of data and control from physical devices to advanced applications.

The perception layer is responsible for collecting data from different sources. This layer includes sensors and actuators that are placed in the environment to gather information about temperature, humidity, light, sound, and other physical parameters. In our case study, the perception level comprises the Smart Meter, Load Control Switch, and Vendor Meter Firmware, which are responsible for collecting real-time data from the grid. Also, it includes the Programmable Communicating Thermostat (PCT) and Remote Terminal Unit (RTU) that facilitate interaction between the control mechanisms and field devices, such as the Circuit Controller.

The processing layer of IoT architecture refers to the software and hardware components that are responsible for collecting, analyzing, and interpreting data from IoT devices. This layer is responsible for receiving raw data from the devices, processing it, and making it available for further analysis or action. This layer includes a variety of technologies and tools, such as data management systems, analytics platforms, and machine learning algorithms. In the examined infrastructure, the processing level includes: (i) the AMI Head-End, which aggregates data from smart meters and field devices, (ii) the Outage Filter that detects and isolates outages, (iii) the Operational Data Store (ODS) which serves as a central repository for storing processed data, (iv) the Meter Data Management (MDM) system that processes data for analytics and reporting, feeding it into downstream systems like Customer Engagement Systems (CES) and D-SCADA, and (v) the CES Controller interfaces with CES Units for demand-side management and power quality monitoring.

Finally, the application layer interacts directly with the end-user. It is responsible for providing user-friendly

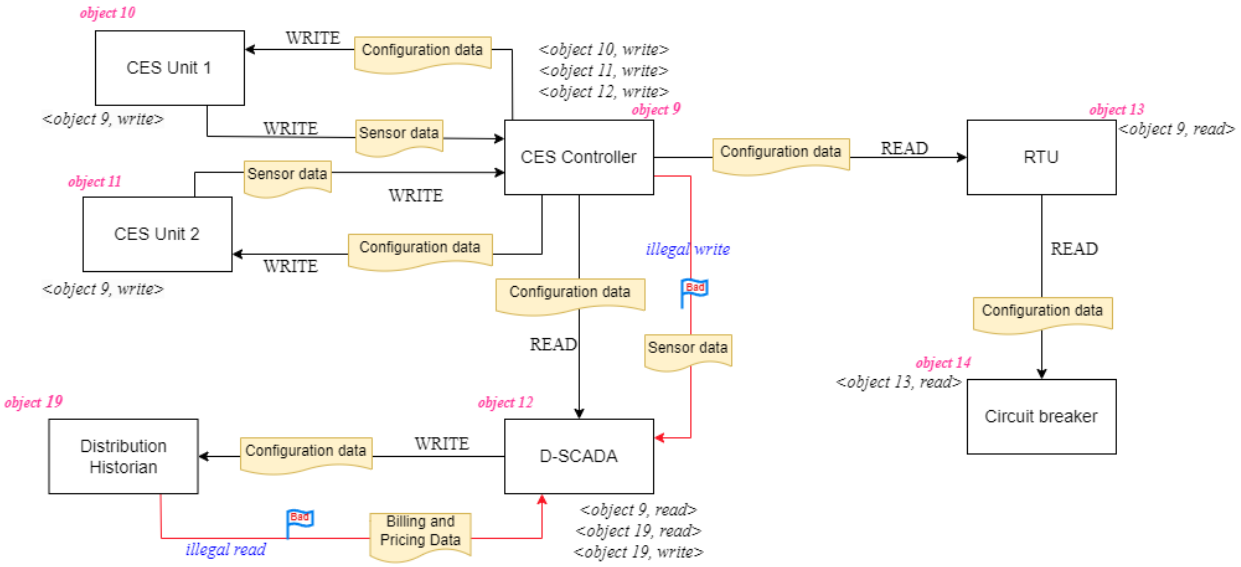


FIGURE 8. Graph representation of the business process: Community Energy Storage (CES) - Energy Dispatch.

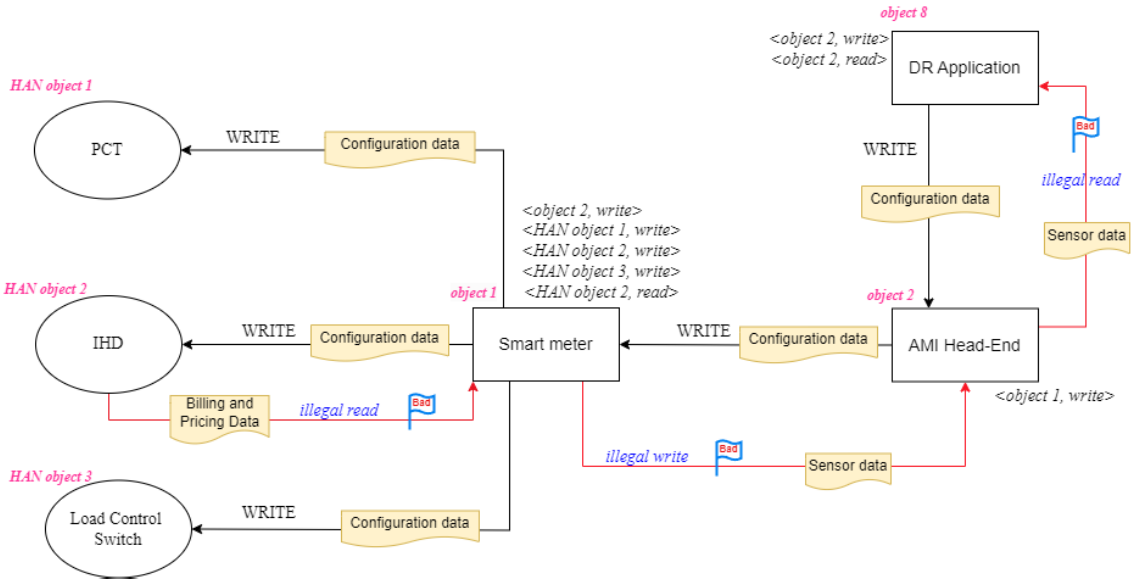


FIGURE 9. Graph representation of the business process: Direct load control event.

interfaces and functionalities that enable users to access and control IoT devices. This layer includes various software and applications such as mobile apps, web portals, and other user interfaces that are designed to interact with the underlying IoT infrastructure. In our case study, the application level integrates various customer-facing and operational systems. The Customer Web Portal which enables end-users to access energy usage insights and billing information. The Demand Response (DR) Application interacts with the Trouble Ticket System (TTS) and Customer Engagement (CE) to optimize energy consumption patterns. The Outage Management System (OMS) collaborates with the DR system to ensure seamless outage resolution. Other key elements include the Distribution Historian, which archives historical operational data, and the In-Home Display (IHD), which provides localized energy consumption information to customers.

1) BULK METER READINGS

This procedure collects and transmits bulk data from smart meters. Every 15 minutes, the smart meter records energy usage, and every 4 hours, this information is sent to the AMI Head-End system. Once a day, the AMI Head-End system forwards these bulk readings in batches to the Meter Data Management (MDM) system. The MDM is tasked

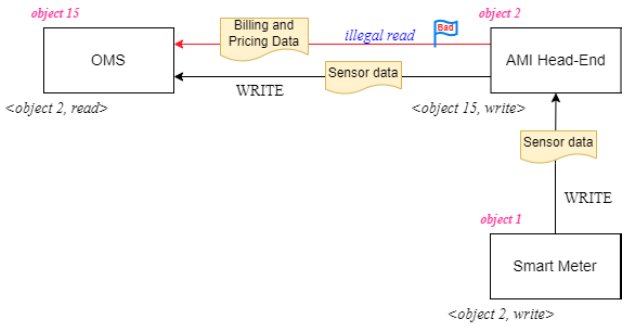


FIGURE 10. Graph representation of the business process: Outage Management System Poll - Multicast.

with validating, estimating, adjusting, and detecting any missing readings from meters. After processing, the MDM relays the readings to the CIS and ODS systems, with ODS sending this data to the Customer Web Portal. In detail, each smart meter is equipped with a component called Meter Metrology Board, which is responsible for measuring and storing electrical consumption at 15-minute intervals. This data is then transmitted every 4 hours by the smart meter's NIC, which serves as a communication interface between the meter and the AMI Head-End via the AMI network. The AMI Head-End system organizes the data into batches and forwards it to the MDM for processing. After validation or adjustments, MDM sends the processed data to the ODS and CIS systems. The ODS routes the data to the customer portal and continuously monitors incoming data streams and metering events [40]. Fig. 4 illustrates the flow of information in this process.

2) DR HAN PRICING AND EVENT CUSTOMER OPT-OUT

The Control Entity (CE) communicates Demand Response (DR) information with the Customer Information System (CIS) and the DR Application, exchanging details related to energy pricing and billing. Specifically, the CE interacts with these systems to send updated Critical Peak Pricing (CPP) or Direct Load Control (DLC) rates to in-home devices like smart meters, thermostats, and load control switches. Customers have the option to participate in a DLC event or opt out. Their choice is sent from their home device to the AMI Head-End system, which passes it on to the DR Application. Based on the customer's decision, the DR Application notifies the CIS to make any necessary changes to billing related to the DLC program. In addition, the DR Application updates the Customer Web Portal with the latest information on CPP rates and DLC events for enrolled customers [40]. Fig. 5 provides a visual representation of the data flow during the customer's opt-out process in DR pricing and event management.

3) IN-FIELD PROGRAMMING OF SMART METER AND METER FIRMWARE UPGRADE

A Smart Meter contains an optical interface that allows for programming and firmware updates using specialized vendor software. This software simplifies the process of installing new firmware or modifying existing configurations. To carry out the update, an electrician connects his laptop to the Smart Meter via an optical probe and uses the vendor's software to load the new firmware. Once the process is complete, the Smart Meter provides feedback, confirming if the update was applied successfully. This status is then communicated to the AMI Head-End system, which subsequently transfers the information to the ODS [40]. Fig. 6 illustrates the flow of information in this process.

4) METER REMOTE CONNECT/DISCONNECT

This process outlines how communication between the CIS and a Smart Meter is handled through the AMI Head-End system and AMI Network, particularly for remote connection or disconnection of meters. The CIS initiates a command for connecting or disconnecting a meter, which is relayed to the Smart Meter via the AMI Head-End. This system enables remote management of meters, such as connecting when a new resident moves in or disconnecting when someone moves out, reinstating services after payment, or disconnecting due to non-payment, among other scenarios. The Customer Service Representative (CSR) triggers a disconnect request in the CIS. The request is sent to the AMI Head-End, which forwards it over the AMI Network to the relevant meter. Upon receiving the request, the meter's NIC receives, processes and forwards it to the Meter Metrology Board, which controls the internal switch responsible for performing the connect or disconnect action. Once the action is executed, a confirmation is sent back to the AMI Head-End and then subsequently forwarded to both the CIS and the ODS, where the event is recorded [40]. Fig. 7 illustrates the information flows during this remote connection and disconnection procedure.

5) COMMUNITY ENERGY STORAGE (CES) - ENERGY DISPATCH

The CES Controller operates according to a scheduled routine, where it regularly gathers information from CES Units about their energy capacity and availability. Additionally, the controller communicates with the substation's circuit breaker via the main RTU. When energy consumption surpasses a specified threshold, the controller initiates a load-balancing event by assessing the data it has received. It then determines the participation level for each CES Unit and issues relevant commands. The CES Units send confirmation acknowledgments to the CES Controller as soon as they receive these commands. Concurrently, the Distribution SCADA system gathers data for real-time monitoring by continuously querying the CES Controllers. In the end, the collected data are stored in the Distribution Historian [40]. Fig. 8 illustrates how information flows within the CES energy dispatch procedure.

6) DIRECT LOAD CONTROL EVENT

A DR system is responsible for managing programs that directly control the electrical loads of certain devices.

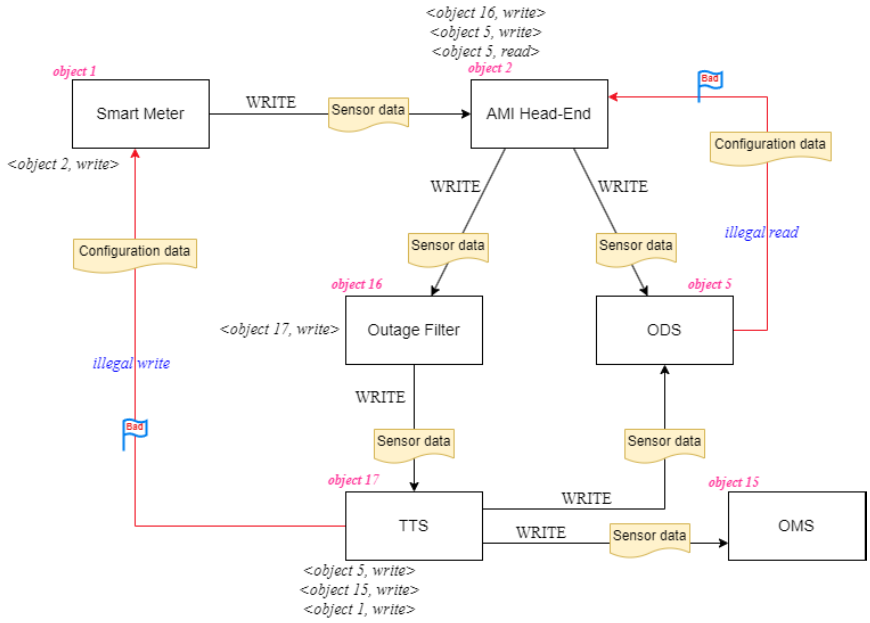


FIGURE 11. Graph representation of the business process: Outage notification.

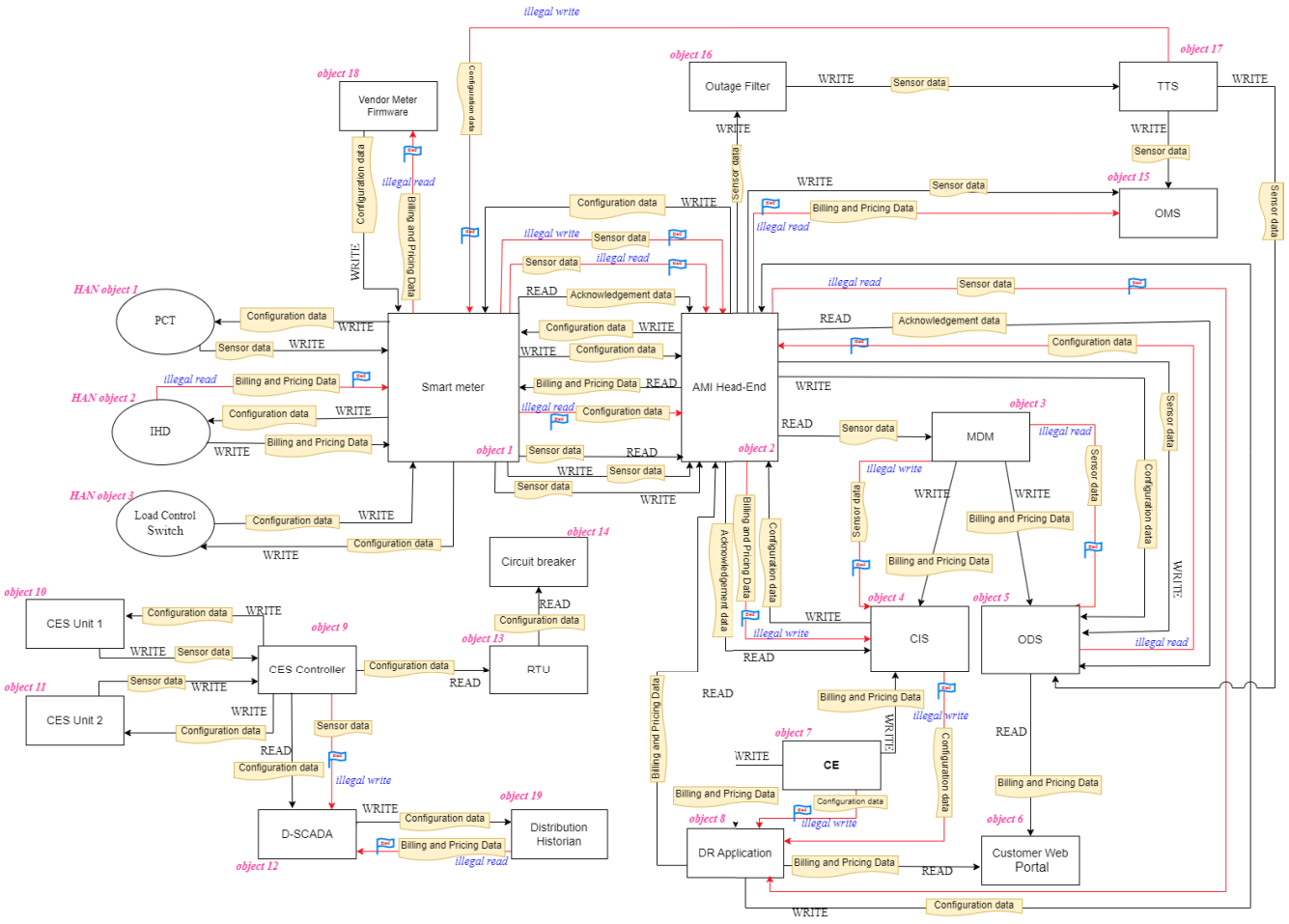


FIGURE 12. Overall graph representation of the business processes within smart grid.

TABLE 8. Estimation of supporting metrics values.

From	To	Data Category	Severity	Operation Type	Operation factor	Transaction Category	Legality
Use Case: Bulk Meter Reading							
Smart Meter	AMI Head-End	Sensor data	4	get	2	Legal Read	1
AMI Head-End	MDM	Sensor data	4	get	2	Legal Read	1
MDM	CIS	Sensor data	4	write	3	Illegal Write	3
MDM	CIS	Billing & Pricing data	3	write	3	Legal Write	1
MDM	ODS	Billing & Pricing data	3	write	3	Legal Write	1
MDM	ODS	Sensor data	4	get	2	Illegal Read	2
ODS	Customer Web Portal	Billing & Pricing data	3	get	2	Legal Read	1
Customer Web Portal	Smart Meter	Billing & Pricing data	3	get	2	Suspicious Read	4
Smart Meter	MDM	Billing & Pricing data	3	write	3	Impossible Write	5

TABLE 9. Calculation of the (a) Illegal Information Flow Likelihood (IIFL), (b) Transaction Impact (TI), and (c) Scaled Transaction Impact (STI).

From	To	IIFL	TI	STI
Use Case: Bulk Meter Reading				
Smart Meter	AMI Head-End	0	8	1.74
AMI Head-End	MDM	0	8	1.74
MDM	CIS	0.5	36	5.19
MDM	CIS	0	9	1.86
MDM	ODS	0	9	1.86
MDM	ODS	0.5	16	2.73
Smart Meter	MDM	1	45	6.30

TABLE 10. Example for the calculation of the Total Transaction Impact (TTI) metric.

From	To	TI
Smart Meter	AMI Head-End	1.74
Smart Meter	AMI Head-End	3.22
Smart Meter	AMI Head-End	2.60
Smart Meter	AMI Head-End	1.00
Smart Meter	AMI Head-End	2.73
Smart Meter	AMI Head-End	5.19
Smart Meter	AMI Head-End	2.23
Smart Meter	AMI Head-End	2.23
AMI Head-End	Smart Meter	1.49
AMI Head-End	Smart Meter	2.60
AMI Head-End	Smart Meter	2.60
	$TTI_{x \to y}$	2.51

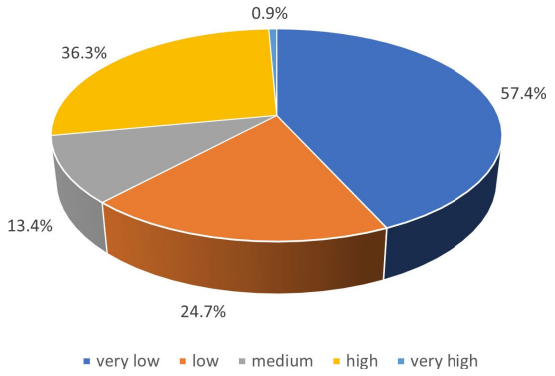


FIGURE 13. Levels of cumulative dependency risk of the dependency chains.

TABLE 11. Calculation of normalized closeness centrality.

Object	Normalized Closeness Centrality
CES Controller	0.75
Smart Meter	0.67
AMI Head-End	0.67
MDM	0.58
D-SCADA	0.55
RTU	0.55
Customer Web Portal	0.52
TTS	0.5
CIS	0.48
ODS	0.48
DR Application	0.48
CES Unit 1	0.46
CES Unit 2	0.46
OMS	0.44
Outage Filter	0.44
Vendor Meter Firmware	0.41
PCT	0.41
IHD	0.41
Load Control Switch	0.41
Circuit breaker	0.38
Distribution Historian	0.38
CE	0.35

TABLE 12. Calculation of bonacich (eigenvector) centrality.

Object	Bonacich (Eigenvector) Centrality
CES Controller	0.549
AMI Head-End	0.471
D-SCADA	0.45
RTU	0.45
Smart Meter	0.404
MDM	0.353
CES Unit 1	0.35
CES Unit 2	0.35
ODS	0.313
CIS	0.29
DR Application	0.27
TTS	0.253
Customer Web Portal	0.235
OMS	0.173
Outage Filter	0.173
Circuit breaker	0.154
Distribution Historian	0.154
Customer Engagement (CE)	0.134
Vendor Meter Firmware	0.096
PCT	0.096
IHD	0.096
Load Control Switch	0.096

It enables communication with consumers via an In-Home Display (IHD), which provides essential information such as energy consumption, pricing details, and signals related to demand response. The system coordinates actions like controlling power to electric vehicles, smart switches, and connected devices within the home network (HAN). The AMI Head-End firstly sends commands over the AMI

TABLE 13. Top 10 dependency chains based on cumulative risk calculation.

1^{st} node	Risk 1	2^{nd} node	Risk 2	3^{rd} node	Risk 3	4^{th} node	Risk 4	5^{th} node	Risk 5	6^{th} node	Risk 6	Cumulative risk
TTS	6.30	Smart Meter	6.30	MDM	1.75	CIS	3.15	DR Application	0.37	AMI Head-End		17.88
TTS	6.30	Smart Meter	6.30	MDM	1.75	CIS	3.15	DR Application				17.5
TTS	6.30	Smart Meter	6.30	MDM	1.75	CIS	3.15	DR Application	0.0	Customer Web Portal		17.5
Customer Web Portal	3.5	Smart Meter	6.30	MDM	1.75	CIS	3.15	DR Application				14.7
Customer Web Portal	3.5	Smart Meter	6.30	MDM	1.75	CIS	3.15	DR Application	0.38	AMI Head-End		14.7
TTS	6.30	Smart Meter	6.30	MDM	1.75	ODS						14.35
TTS	6.30	Smart Meter	6.30	MDM	1.15	ODS	0.0	Customer Web Portal				13.75
TTS	6.30	Smart Meter	6.30	MDM	1.15	ODS						13.75
TTS	6.30	Smart Meter	6.30	MDM	1.15	ODS	0.0	Customer Web Portal	0.0	Smart Meter		13.75
TTS	6.30	Smart Meter	6.30	MDM								12.6

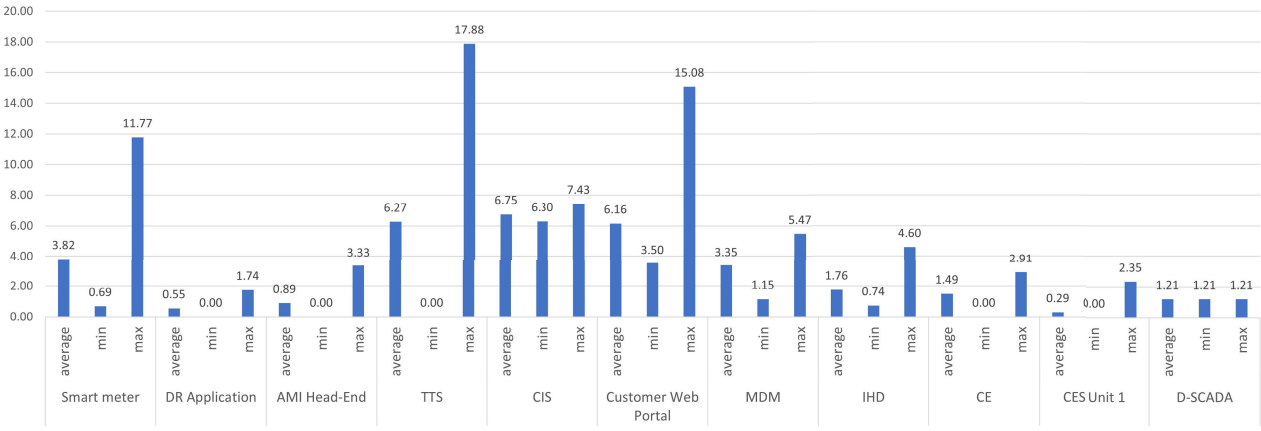


FIGURE 14. Cumulative dependency risk per object.

network to the smart meter once a demand response event is activated by a DR application. Then, it forwards the commands to the appropriate devices within the HAN. These devices may include smart thermostats, load switches, or the IHD, receiving instructions on adjustments to settings like temperature, pricing, and other important notifications [40]. The information flows during a direct load control event are outlined in Fig. 9.

7) OUTAGE MANAGEMENT SYSTEM POLL - MULTICAST

The polling procedure managed by the Outage Management System (OMS) to confirm the status of an ongoing outage. The OMS initiates a command to smart meters to check whether the outage persists. This polling request can be triggered automatically by the system or manually by an operator. Once initiated, the OMS issues a multicast command through the AMI to multiple smart meters simultaneously. Upon receiving the command, the meter's NIC forwards it to the metrology board, which measures the voltage level. The resulting voltage data are then transmitted back through the AMI network to the OMS via the AMI system. This process helps determine the current status of the outage [40]. Fig. 10 presents the information flows of this process.

8) OUTAGE NOTIFICATION

Most AMI systems are equipped with a "last gasp" messaging capability that automatically alerts utility providers when a power outage occurs at a customer's location. This automated notification functions similarly to a manual outage report from the customer. By leveraging this technology, utility companies can enhance their OMS and dispatcher capabilities, allowing for more efficient responses to widespread power disruptions. The "last gasp" message is triggered directly by the smart meter's NIC when it detects a sustained power loss beyond a predetermined amount of time. Upon detection, the NIC independently sends a "last gasp" signal to the utility without requiring any external input. This signal travels through the AMI network to the AMI Head-End system, which then forwards the message to the ODS and the Outage Filter. After filtering, the message is passed to the Trouble Ticket System (TTS), which updates both the OMS and ODS. The OMS then logs the outage event and

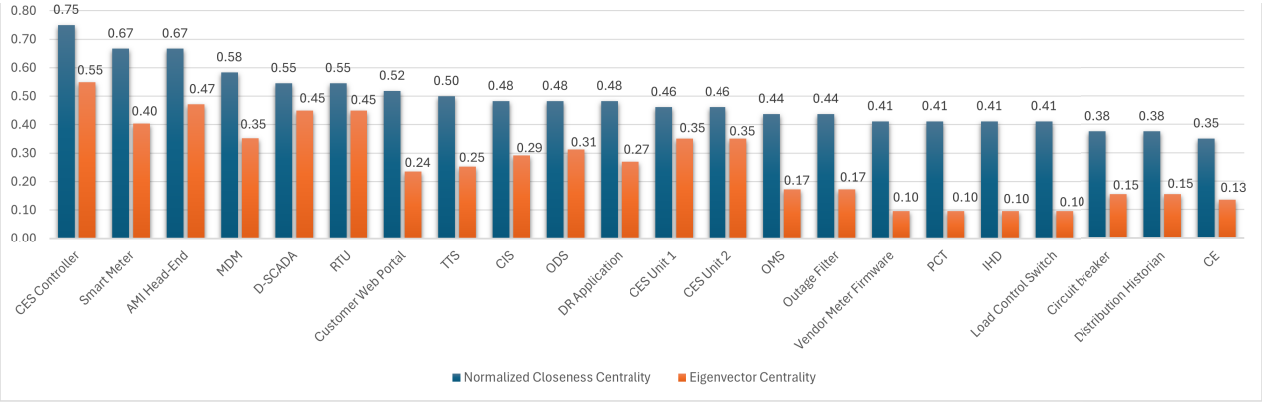


FIGURE 15. Normalized closeness vs bonacich (eigenvector).

synchronizes updates across the TTS and ODS systems [40]. Fig. 11 illustrates the flow of information during this process.

Fig. 12 depicts the overall graph of all the aforementioned business processes. This figure serves as an alternative representation of Fig. 3, which illustrates the system architecture diagram highlighting the components involved in the selected use cases. While Fig. 3 provides a structural overview of the components of the system and their interactions, Fig. 12 offers a more process-oriented perspective, focusing on the information flows across the business processes. These figures complement each other, offering a holistic view of the system functionality and the integration of its components.

B. DEPENDENCY CHAIN ANALYSIS

In this section, we first calculate the first-order dependencies (*see steps 1 to 4 of phase 2*). Subsequently, we calculate the n-order dependencies (*see step 5 of phase 2*).

1) FIRST-ORDER DEPENDENCIES

In order to calculate the first-order dependencies, we should firstly calculate the metrics of Severity, Operation factor, as well as Legality for each information flow. The values of these metrics are comprehensively presented on Appendix A (*see Table 15*), grouped based on the use case that the information flows were associated with. Table 8 presents a part of these calculations. After estimating these values, we calculate the (a) illegal information flow likelihood (IIFL), (b) the transaction impact (TI), and consequently (c) the scaled transaction impact (STI). The results of these calculations are comprehensively presented on Appendix B (*see Table 16*). Table 9 presents an indicative part of these calculations.

Finally, we calculate the Total Transaction Impact ($TTI_{x\to y}$) for all the information flows. Table 10 provides an example on how we compute TTI of the information flows between a smart meter and the AMI head-end. On Appendix C (*see Table 17*) we present the overall results for this metric. We focus on the transaction between objects, regardless of its starting or ending point; in other words, we focus on the pair of objects that participate in this transaction.

2) MULTI-ORDER DEPENDENCIES

Our implementation is based on the CIDA tool [39]. We modified and expanded the functionalities of this tool in order to assess the information flows of business processes within an industrial environment. The CIDA tool generated all the dependencies for each object. Table 13 presents the top 10 dependency chains based on cumulative risk calculation.

In Fig. 13 we present a pie including the percentages of how the cumulative risks of dependency chains are allocated. We notice that the dependency chains are mainly distributed in two diverse groups: those with either very low or high cumulative dependency risk.

Fig. 14 presents the min, max, and average cumulative dependency risk for each object. We observe that the higher cumulative dependency risk belongs to TTS. As a result, dependency chains involving TTS also have an increased risk.

C. CENTRALITY METRICS ANALYSIS

In our approach, we calculate two graph centrality metrics: (1) normalized closeness, and (2) Bonacich (eigenvector). These metrics are important for our analysis, since they can indicate the importance of objects in a dependency risk graph.

1) NORMALIZED CLOSENESS CENTRALITY ANALYSIS

Table 11 presents the normalized closeness centrality values for each object. Objects with higher values indicate that they are closer to many others in the network. This means that such objects may play a critical role in the flow of information within the system. In our case, the CES Controller stands out with a centrality of 0.75, followed closely by the Smart Meter and the AMI Head-End with centrality values of 0.67. Objects with moderate centrality values (around 0.5 to 0.6) are also important, but not as critical as the highest centrality objects. These objects, such as MDM, D-SCADA, and RTU, still play an important role in the information flow control within the network. Finally, objects with lower centrality values

TABLE 14. Comparative analysis of the proposed methodology and existing approaches.

Paper	Domain		Inclusion of Metrics					Risk estimation method		Type of analysis		Evaluation Methods		
	Access control	Information flow control	Risk propagation	Illegal flow detection	Centralities	Threat assessment	Action Severity	Fuzzy Logic	Mathematical Functions	Static	Real-time	Simulation	Real-world case studies	Not discussed
[18]	✓					✓	✓		✓	✓				✓
[19]	✓					✓	✓	✓		✓		✓		
[20]	✓		✓	✓		✓	✓		✓	✓	✓			✓
[21]	✓					✓			✓		✓	✓		
[22]	✓					✓	✓	✓	✓		✓	✓		
[23]	✓					✓	✓	✓	✓		✓	✓		
[24]	✓					✓	✓		✓		✓	✓		
[25]	✓					✓	✓		✓	✓				✓
Proposed methodology	✓	✓	✓	✓	✓	✓	✓		✓	✓		✓		

(below 0.5) are less central to the network. These objects may not directly influence the flow of information to the same extent as higher centrality nodes. Indicative examples include the Circuit breaker and Customer Engagement (CE) with centrality values of 0.38 and 0.35 respectively.

2) BONACICH (EIGENVECTOR) CENTRALITY ANALYSIS

Table 12 presents the calculation of Bonacich (eigenvector) centrality metric per object. Objects with higher Bonacich centrality values indicate that they are more influential within the network, considering both their direct connections and the centrality of their connections. In our case, the CES Controller stands out with the highest centrality value of 0.549, followed by the AMI Head-End, and the D-SCADA. Moreover, objects with high Bonacich centrality values can be considered key players in the network due to their influence over other influential objects. These objects, such as the CES Controller, are important for the propagation of illegal information flows throughout the network.

3) NORMALIZED CLOSENESS VS BONACICH (EIGENVECTOR)

Comparing the Bonacich centrality and Normalized Closeness centrality values provides a comprehensive understanding of the infrastructure's structure and the roles of individual objects in facilitating illegal information flows. Fig. 15 presents a bar chart which contrasts two centrality metrics for each object: Normalized Closeness centrality, represented by blue bars, and Bonacich centrality, represented by orange bars. Below, we outline our findings:

- **Consistency in Centrality Rankings:** In many cases, objects with high Bonacich centrality also exhibit high Normalized Closeness Centrality, indicating consistency in their importance and influence within the network. For example, CES Controller and AMI Head-End are among the top objects in both centrality measures.
- **Identification of Key Players:** Objects with consistently high centrality values in both metrics, such as the CES Controller, AMI Head-End, and D-SCADA, can be considered key players in the network. These objects not only have many direct connections but also exert influence over other influential objects, making them crucial for the propagation of illegal information flows.
- **Centrality Value Distribution:** The distribution of the values shows a clear differentiation between the more central and peripheral objects, which can be crucial for identifying potential points of vulnerability or influence.

VI. DISCUSSION

Research in cyber-physical systems (CPS), industrial control systems (ICS), and Industrial Internet of Things (IIoT) environments has thoroughly explored methodologies such as graph modeling of business processes, process-based risk assessment, and anomaly detection. These approaches have demonstrated effectiveness in specific domains, including identifying cascading failures [34], modeling attack paths for vulnerability analysis [48], enhancing resilience through information flow analysis [49], addressing IoT and IIoT challenges in risk modeling [50], and mitigating cyber risks with probabilistic attack graphs [51], [52]. Despite their contributions, they primarily emphasize risk assessment and anomaly detection rather than addressing the dynamic challenges of risk-based access control and information flow control.

In IIoT environments, decentralized architectures and complex information flows require robust mechanisms to mitigate illegal data propagation and enforce precise access control policies. Existing methodologies, such as information flow tracking for securing CPS against hardware Trojans [53]

TABLE 15. Detailed calculation of supporting metrics.

From	To	Data Category	Severity	Operation Type	Operation factor	Transaction Category	Legality
Use Case: Bulk Meter Reading							
Smart Meter	AMI Head-End	Sensor data	4	get	2	Legal Read	1
AMI Head-End	MDM	Sensor data	4	get	2	Legal Read	1
MDM	CIS	Sensor data	4	write	3	Illegal Write	3
MDM	CIS	Billing & Pricing data	3	write	3	Legal Write	1
MDM	ODS	Billing & Pricing data	3	write	3	Legal Write	1
MDM	ODS	Sensor data	4	get	2	Illegal Read	2
ODS	Customer Web Portal	Billing & Pricing data	3	get	2	Legal Read	1
Customer Web Portal	Smart Meter	Billing & Pricing data	3	get	2	Suspicious Read	4
Smart Meter	MDM	Billing & Pricing data	3	write	3	Impossible Write	5
Use Case: DR HAN Pricing							
PCT	Smart Meter	Sensor data	4	write	3	Legal Write	1
IHD	Smart Meter	Billing & Pricing data	3	write	3	Legal Write	1
Load Control Switch	Smart Meter	Configuration data	5	write	3	Legal Write	1
Smart Meter	AMI Head-End	Configuration data	5	get	2	Illegal Read	2
AMI Head-End	Smart Meter	Billing & Pricing data	3	get	2	Legal Read	1
CE	CIS	Billing & Pricing data	3	write	3	Legal Write	1
CE	DR Application	Billing & Pricing data	3	write	3	Legal Write	1
CE	DR Application	Configuration data	5	write	3	Illegal Write	3
CIS	DR Application	Configuration data	5	write	3	Illegal Write	3
DR Application	AMI Head-End	Billing & Pricing data	3	get	2	Legal Read	1
DR Application	Customer Web Portal	Billing & Pricing data	3	get	2	Legal Read	1
Use Case: In-Field Programming of Smart Meter							
Smart Meter	Vendor Meter Firmware	Billing & Pricing data	3	get	2	Illegal Read	2
Vendor Meter Firmware	Smart Meter	Configuration data	5	write	3	Legal Write	1
Smart Meter	AMI Head-End	Configuration data	5	write	3	Legal Write	1
AMI Head-End	ODS	Configuration data	5	write	3	Legal Write	1
Use Case: Meter Remote Connect / Disconnect							
Smart Meter	AMI Head-End	Acknowledgement data	1	get	2	Legal Read	1
Smart Meter	AMI Head-End	Sensor data	4	get	2	Illegal Read	2
AMI Head-End	Smart Meter	Configuration data	5	write	3	Legal Write	1
AMI Head-End	CIS	Acknowledgement data	1	get	2	Legal Read	1
AMI Head-End	CIS	Billing & Pricing data	3	write	3	Illegal Write	3
AMI Head-End	ODS	Acknowledgement data	1	get	2	Legal Read	1
CIS	AMI Head-End	Configuration data	5	write	3	Legal Write	1
Use Case: CES Energy dispatch							
CES Unit 1	CES Controller	Sensor data	4	write	3	Legal Write	1
CES Unit 2	CES Controller	Sensor data	4	write	3	Legal Write	1
CES Controller	CES Unit 1	Configuration data	5	write	3	Legal Write	1
CES Controller	CES Unit 2	Configuration data	5	write	3	Legal Write	1
CES Controller	D-SCADA	Configuration data	5	get	2	Legal Read	1
CES Controller	D-SCADA	Sensor data	4	write	3	Illegal Write	3
CES Controller	RTU	Configuration data	5	get	2	Legal Read	1
RTU	Circuit breaker	Configuration data	5	get	2	Legal Read	1
D-SCADA	Distribution Historian	Configuration data	5	write	3	Legal Write	1
Distribution Historian	D-SCADA	Billing & Pricing data	3	get	2	Illegal Read	2
Use Case: Direct Load Control Event							
Smart Meter	PCT	Configuration data	5	write	3	Legal Write	1
Smart Meter	IHD	Configuration data	5	write	3	Legal Write	1
Smart Meter	Load Control Switch	Configuration	5	write	3	Legal Write	1
Smart Meter	AMI Head-End	Sensor data	4	write	3	Illegal Write	3
IHD	Smart Meter	Billing & Pricing	3	get	2	Illegal Read	2
AMI Head-End	Smart Meter	Configuration data	5	write	3	Legal Write	1
AMI Head-End	DR Application	Sensor data	4	get	2	Illegal Read	2
DR Application	AMI Head-End	Configuration data	5	write	3	Legal Write	1
Use Case: Outage Management System Poll							
Smart Meter	AMI Head-End	Sensor data	4	write	3	Legal Write	1
AMI Head-End	OMS	Sensor data	4	write	3	Legal Write	1
AMI Head-End	OMS	Billing & Pricing data	3	get	2	Illegal Read	2
Use Case: Outage Notification							
Smart Meter	AMI Head-End	Sensor	4	write	3	Legal Write	1
AMI Head-End		Sensor data	4	write	3	Legal Write	1
AMI Head-End	Outage Filter	Sensor data	4	write	3	Legal Write	1
ODS	AMI Head-End	Configuration data	5	get	2	Illegal Read	2
Outage Filter	TTS	Sensor data	4	write	3	Legal Write	1
TTS	Smart Meter	Configuration data	5	write	3	Illegal Write	3
TTS	ODS	Sensor data	4	write	3	Legal Write	1
TTS	OMS	Sensor data	4	write	3	Legal Write	1

TABLE 16. Detailed calculation of (a) Illegal Information Flow Likelihood (IIFL), (b) Transaction Impact (TI), and (c) Scaled Transaction Impact (STI).

From	To	IIFL	TI	STI
Use Case: Bulk Meter Reading				
Smart Meter	AMI Head-End	0	8	1.74
AMI Head-End	MDM	0	8	1.74
MDM	CIS	0.5	36	5.19
MDM	CIS	0	9	1.86
MDM	ODS	0	9	1.86
MDM	ODS	0.5	16	2.73
ODS	Customer Web Portal	0	6	1.49
Customer Web Portal	Smart Meter	1	24	3.71
Smart Meter	MDM	1	45	6.30
Use Case: DR HAN Pricing				
PCT	Smart Meter	0	12	2.23
IHD	Smart Meter	0	9	1.86
Load Control Switch	Smart Meter	0	15	2.60
Smart Meter	AMI Head-End	0.5	20	3.22
AMI Head-End	Smart Meter	0	6	1.49
CE	CIS	0	9	1.86
CE	DR Application	0	9	1.86
CE	DR Application	0.5	45	6.30
CIS	DR Application	1	45	6.30
DR Application	AMI Head-End	0	6	1.49
DR Application	Customer Web Portal	0	6	1.49
Use Case: In-Field Programming of Smart Meter				
Smart Meter	Vendor Meter Firmware	0.5	12	2.23
Vendor Meter Firmware	Smart Meter	0	15	2.60
Smart Meter	AMI Head-End	0	15	2.60
AMI Head-End	ODS	0	15	2.60
Use Case: Meter Remote Connect / Disconnect				
Smart Meter	AMI Head-End	0	2	1.00
Smart Meter	AMI Head-End	0.33	16	2.73
AMI Head-End	Smart Meter	0	15	2.60
AMI Head-End	CIS	0	2	1.00
AMI Head-End	CIS	0.33	27	4.08
AMI Head-End	ODS	0	2	1.00
CIS	AMI Head-End	0	15	2.60
Use Case: CES Energy dispatch				
CES Unit 1	CES Controller	0	12	2.23
CES Unit 2	CES Controller	0	12	2.23
CES Controller	CES Unit 1	0	15	2.60
CES Controller	CES Unit 2	0	15	2.60
CES Controller	D-SCADA	0	10	1.99
CES Controller	D-SCADA	0.5	36	5.19
CES Controller	RTU	0	10	1.99
RTU	Circuit breaker	0	10	1.99
D-SCADA	Distribution Historian	0	15	2.60
Distribution Historian	D-SCADA	0.5	12	2.23
Use Case: Direct Load Control Event				
Smart Meter	PCT	0	15	2.60
Smart Meter	IHD	0	15	2.60
Smart Meter	Load Control Switch	0	15	2.60
Smart Meter	AMI Head-End	0.5	36	5.19
IHD	Smart Meter	0.5	12	2.23
AMI Head-End	Smart Meter	0	15	2.60
AMI Head-End	DR Application	0.5	16	2.73
DR Application	AMI Head-End	0	15	2.60
Use Case: Outage Management System Poll				
Smart Meter	AMI Head-End	0	12	2.23
AMI Head-End	OMS	0	12	2.23
AMI Head-End	OMS	0.5	12	2.23
Use Case: Outage Notification				
Smart Meter	AMI Head-End	0	12	2.23
AMI Head-End	ODS	0	12	2.23
AMI Head-End	Outage Filter	0	12	2.23
ODS	AMI Head-End	0.5	20	3.22
Outage Filter	TTS	0	12	2.23
TTS	Smart Meter	1	45	6.30
TTS	ODS	0	12	2.23
TTS	OMS	0	12	2.23

TABLE 17. Detailed calculation of the Total Transaction Impact.

From	To	$TTI_{x \to y}$
CIS	DR Application	6.30
TTS	Smart Meter	6.30
Smart Meter	MDM	6.30
CE	DR Application	4.08
Customer Web Portal	Smart Meter	3.71
CES Controller	D-SCADA	3.59
MDM	CIS	3.53
Smart Meter	Load Control Switch	2.60
AMI Head-End	CIS	2.56
Smart Meter	AMI Head-End	2.51
CES Controller	CES Unit 2	2.42
CES Unit 1	CES Controller	2.42
Distribution Historian	D-SCADA	2.42
IHD	Smart Meter	2.42
Smart Meter	PCT	2.42
Smart Meter	Vendor Meter Firmware	2.42
MDM	ODS	2.29
AMI Head-End	DR Application	2.27
ODS	AMI Head-End	2.26
AMI Head-End	OMS	2.23
AMI Head-End	Outage Filter	2.23
Outage Filter	TTS	2.23
TTS	ODS	2.23
TTS	OMS	2.23
CES Controller	RTU	1.99
RTU	Circuit breaker	1.99
CE	CIS	1.86
IHD	Smart Meter	1.86
AMI Head-End	MDM	1.74
DR Application	Customer Web Portal	1.49

or using attack graphs for security hardening [52], offer valuable insights but lack an integrated framework for dynamically managing risks and controlling information flows. Our study bridges this gap by introducing a unified methodology that combines advanced dependency analysis, centrality metrics, and graph-based risk propagation tailored for IIoT environments. This approach provides a comprehensive mechanism to detect and mitigate risks associated with illegal information flows, extending beyond traditional security measures.

While acknowledging the contributions of these approaches, we exclude them from direct comparison as their primary focus diverges from risk-based access control and information flow-centric security frameworks, which are central to our study. This section presents a detailed comparison of our methodology with the existing studies discussed in Section II. This analysis evaluates key criteria such as domain focus, specific metrics, risk estimation methods, type of analysis, and evaluation techniques, highlighting the novelty and contribution of our methodology within the context of access control and information flow control. To facilitate our comparative analysis, we define five criteria.

Criterion 1: Domain. This criterion evaluates whether studies address *Access Control*, ensuring only authorized entities interact with resources (e.g., RBAC, DAC), or/and *Information Flow Control*, which manages how information flows to prevent exposure, critical in IIoT environments.

Criterion 2: Inclusion of Metrics. This criterion assesses the analytical depth of the methodologies:

- *Risk Propagation.* Evaluates how risk could spread from one component to another within a network, indicating potential impacts and vulnerabilities.
- *Illegal Flow Detection.* Identifies unauthorized information flows in order to detect potential security breaches or vulnerabilities.
- *Centralities.* Highlights key network nodes that need enhanced security controls.
- *Action severity.* Evaluates transaction criticality and potential impact.
- *Threat assessment.* Uses prior knowledge of vulnerabilities or threats for the evaluation.

Criterion 3: Risk estimation method. This criterion evaluates the technique used to calculate risk. Risk estimation methods can be either (i) *Fuzzy logic*, a flexible approach that can assess risk by considering multiple factors, each with varying degrees of influence, or (ii) *Mathematical functions* which rely on traditional mathematical models and statistical functions to calculate risk levels.

Criterion 4: Type of analysis. This criterion differentiates methodologies based on whether they perform one-time evaluations (*static analysis*) or continuous monitoring for dynamic risk adaptation, essential in IIoT (*real-time analysis*).

Criterion 5: Evaluation Methods. This criterion assesses the techniques used to validate a methodology:

- *Simulation.* Approaches are evaluated on virtual scenarios to test their performance under controlled conditions.
- *Real-world case studies.* Approaches are evaluated on real-world scenarios to assess their practical effectiveness and scalability, providing a stronger evidence of their applicability in real operational environments.
- *Not discussed.* Approaches do not provide information regarding their evaluation.

Table 14 indicates that the proposed methodology differs from existing research in several ways. Notably, it is the only approach that incorporates centrality metrics, which significantly improves its performance on risk analysis. In environments where maintaining information integrity and confidentiality is crucial, it is important to effectively assess the influence and criticality of nodes within the network.

Moreover, unlike other studies that primarily focus on access control, the proposed methodology comprehensively addresses both the domains of information flow control and access control. Our approach aligns with most existing works, except for [21]. It includes action severity and threat assessment features, which drive to a more detailed evaluation of the risks associated with access requests. This also provides a more accurate estimation of potential impacts arising from security events. The features of risk propagation and detection of illegal flows within the network are only addressed both by the proposed methodology and the [20].

However, the proposed work shares limitations with existing research. Similar to the other works, it relies on static analysis, evaluating risks one time rather than continuously in real-time. Its static nature may limit its applicability in dynamic environments, such as smart grids, where real-time responsiveness is essential to rapidly adapt to evolving risks.

VII. CONCLUSION

In this paper, we extended a prior work of a risk-based information flow control methodology in IIoT environments. Specifically, in our current work we incorporated formulae for the calculation of multi-order dependencies. An n-order dependency exists when the relationship between two objects involves intermediate objects. Understanding n-order dependencies is essential for comprehensive risk analysis. Risks can propagate through multiple objects and pathways. The analysis of these multi-order dependencies helps reveal hidden vulnerabilities that may not be obvious when focusing only on direct relationships.

We also estimated two graph centrality metrics: Normalized Closeness and Bonacich (Eigenvector). While Bonacich and Normalized Closeness centralities both assess the importance of objects in a network, they focus on different aspects. Bonacich centrality considers both an object's direct connections and the centrality of its neighbors, whereas Normalized Closeness centrality focuses solely on the distance between an object and all other objects in the network. Analyzing both metrics gives a broader view of how objects participate in illegal information flows. For instance, if an object has much lower centrality than expected given its importance, it may indicate areas that need improvements to enhance the overall efficiency and robustness. On the other hand, objects with high centrality play a key role in information flow. Comparing these values helps security experts to gain valuable insights regarding the key objects, vulnerability, and opportunities for optimizing the network's security and efficiency.

Our study is subject to several limitations. Firstly, due to the focus on critical infrastructures, particularly smart grids, real-world datasets are not publicly available [35]. Consequently, we relied on descriptions of real-world business processes to model them as directed graphs and create data records. This approach, while practical, may not fully capture the complexity and variability of real-world scenarios. Furthermore, the chosen centrality metrics, although effective, may not encompass all aspects of node importance and influence in the network. Other metrics or a combination of multiple metrics might provide a more comprehensive analysis. Finally, the proposed methodology performs static analysis, since it evaluates risk one time, which limits its ability to respond dynamically to changes in IIoT environments.

Building on our findings, future work can focus on several key areas to enhance the robustness and applicability of our methodology. To overcome the limitations of dataset

constraints, future research should obtain access to more comprehensive and diverse datasets. Collaborations with industry partners could provide valuable real-world data, enhancing the validity and applicability of our approach. Expanding the application to different IIoT environments, such as healthcare and transportation, can help validate its flexibility and effectiveness across various domains. Exploring other centrality metrics, e.g. Betweenness Centrality, can provide a more thorough understanding of the roles of different objects within the network. To address the static nature, we will focus on adapting the methodology to support real-time analysis. This can be achieved by transforming the current risk calculation formulae. Finally, extending the approach to efficiently detect illegal information flows in covert channel attacks is an interesting avenue. This involves developing new detection and prevention methods tailored to such sophisticated threats.

APPENDIX A
CALCULATION OF SUPPORTING METRICS

Table 15 depicts the detailed calculation of the metrics of Severity, Operation factor, as well as Legality for each information flow. The values of these metrics are grouped based on the use case that the information flows were associated with.

APPENDIX B
CALCULATION OF TRANSACTION IMPACT PARAMETERS

Table 16 calculates the parameters of (a) illegal information flow likelihood (IIFL), (b) the transaction impact (TI), and consequently (c) the scaled transaction impact (STI) for all the use cases.

APPENDIX C
CALCULATION OF TOTAL TRANSACTION IMPACT

Table 17 depicts the detailed calculation of the Total Transaction Impact ($TTI_{x \to y}$) for all the information flows.

REFERENCES

[1] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4724–4734, Nov. 2018.

[2] M. Enisa, "Good practices for security of Internet of Things in the context of smart manufacturing," in *European Union Agency for Network and Information Security (ENISA)*. Crete, Greece, 2018.

[3] S. Nakamura, T. Enokido, L. Barolli, and M. Takizawa, "Capability-based information flow control model in the IoT," in *Proc. Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput.* Cham, Switzerland: Springer, Jun. 2019, pp. 63–71.

[4] P. Samarati and S. C. D. Vimercati, "Access control: Policies, models, and mechanisms," in *International School on Foundations of Security Analysis and Design*. Cham, Switzerland: Springer, 2000, pp. 137–196.

[5] S. Salonikias, A. Gouglidis, I. Mavridis, and D. Gritzalis, "Access control in the industrial Internet of Things," in *Security and Privacy Trends in the Industrial Internet of Things*. Cham, Switzerland: Springer, 2019, pp. 95–114.

[6] M. Jaume, V. V. T. Tong, and L. Mé, "Flow based interpretation of access control: Detection of illegal information flows," in *Proc. 7th Int. Conf. Inf. Syst. Secur.*, Kolkata, India, Jan. 2011, pp. 72–86.

[7] A. Anagnostopoulou, I. Mavridis, and D. Gritzalis, "Risk-based illegal information flow detection in the IIoT," in *Proc. 20th Int. Conf. Secur. Cryptography*, 2023, pp. 377–384.

[8] H. A. Daniel and S. Andrei, "A perspective on information-flow control," in *Software Safety and Security*. Amsterdam, The, Netherlands: IOS Press, Jan. 2012, pp. 319–347.

[9] W. Hu, A. Ardeshiricham, and R. Kastner, "Hardware information flow tracking," *ACM Comput. Surv.*, vol. 54, no. 4, pp. 1–39, May 2022.

[10] J. Shin, H. Zhang, J. Lee, I. Heo, Y.-Y. Chen, R. Lee, and Y. Paek, "A hardware-based technique for efficient implicit information flow tracking," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, Nov. 2016, pp. 1–7.

[11] C. Deutschbein, A. Meza, F. Restuccia, R. Kastner, and C. Sturton, "Isadora: Automated information flow property generation for hardware designs," in *Proc. 5th Workshop Attacks Solutions Hardw. Secur.*, Nov. 2021, pp. 5–15.

[12] D. Zhang, Y. Wang, G. E. Suh, and A. C. Myers, "A hardware design language for timing-sensitive information-flow security," *ACM SIGPLAN Notices*, vol. 50, no. 4, pp. 503–516, May 2015.

[13] N. Broberg, B. v. Delft, and D. Sands, "Paragon for practical programming with information-flow control," in *Proc. 11th Asian Symp. Program. Lang. Syst.* Cham, Switzerland: Springer, Jan. 2013, pp. 217–232.

[14] A. Sabelfeld and A. C. Myers, "Language-based information-flow security," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 1, pp. 5–19, Jan. 2003.

[15] A. C. Myers, "JFlow: Practical mostly-static information flow control," in *Proc. 26th ACM SIGPLAN-SIGACT Symp. Princ. Program. Lang.*, Jan. 1999, pp. 228–241.

[16] N. Halbwachs, P. Caspi, P. Raymond, and D. Pilaud, "The synchronous data flow programming language LUSTRE," *Proc. IEEE*, vol. 79, no. 9, pp. 1305–1320, Sep. 1991.

[17] L. Zheng and A. C. Myers, "Dynamic security labels and static information flow control," *Int. J. Inf. Secur.*, vol. 6, nos. 2–3, pp. 67–84, Mar. 2007.

[18] H. Khambhammettu, S. Boulares, K. Adi, and L. Logrippo, "A framework for risk assessment in access control systems," *Comput. Secur.*, vol. 39, pp. 86–103, Nov. 2013.

[19] Q. Ni, E. Bertino, and J. Lobo, "Risk-based access control systems built on fuzzy inferences," in *Proc. 5th ACM Symp. Inf., Comput. Commun. Secur.*, Apr. 2010, pp. 250–260.

[20] L. Zhang, A. Brodsky, and S. Jajodia, "Toward information sharing: Benefit and risk access control (BARAC)," in *Proc. 7th IEEE Int. Workshop Policies Distrib. Syst. Netw. (POLICY)*, Jun. 2006, p. 9.

[21] R. A. Shaikh, K. Adi, and L. Logrippo, "Dynamic risk-based decision methods for access control systems," *Comput. Secur.*, vol. 31, no. 4, pp. 447–464, Jun. 2012.

[22] H. F. Atlam and G. B. Wills, "An efficient security risk estimation technique for risk-based access control model for IoT," *Internet Things*, vol. 6, Jun. 2019, Art. no. 100052.

[23] H. F. Atlam and G. B. Wills, "ANFIS for risk estimation in risk-based access control model for smart homes," *Multimedia Tools Appl.*, vol. 82, no. 12, pp. 18269–18298, May 2023.

[24] M. B. Aliyu, M. Garba, D. Gabi, H. U. Suru, and M. S. Argungu, "Dynamic access control at the network edge using an adaptive risk-based access control system (ad-RACs)," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 3, pp. 1–21, 2024.

[25] L. Chen and J. Crampton, "Risk-aware role-based access control," in *Proc. 7th Int. Workshop Secur. Trust Manag.*, Copenhagen, Denmark, Jan. 2012, pp. 140–156.

[26] P. Kotzanikolaou, M. Theoharidou, and D. Gritzalis, "Cascading effects of common-cause failures in critical infrastructures," in *Proc. Int. Conf. Crit. Infrastruct. Protection*, Washington, DC, USA, Jan. 2013, pp. 171–182, doi: 10.1007/978-3-642-45330-4_12.

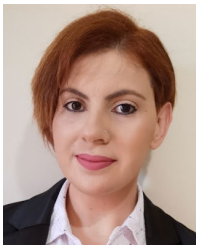
[27] P. Kotzanikolaou, M. Theoharidou, and D. Gritzalis, "Assessing n-order dependencies between critical infrastructures," *Int. J. Crit. Infrastructures*, vol. 9, nos. 1–2, p. 93, 2013, doi: 10.1504/ijcis.2013.051606.

[28] R. S. Sandhu, "Lattice-based access control models," *Computer*, vol. 26, no. 11, pp. 9–19, Nov. 1993.

[29] D. E. Denning, "A lattice model of secure information flow," *Commun. ACM*, vol. 19, no. 5, pp. 236–243, May 1976.

[30] D. E. Bell and L. J. LaPadula, "Secure computer systems: A mathematical model, volume II," *J. Comput. Secur.*, vol. 4, nos. 2–3, pp. 229–263, 1996.

[31] D. D. Downs, J. R. Rub, K. C. Kung, and C. S. Jordan, "Issues in discretionary access control," in *Proc. IEEE Symp. Secur. Privacy*, Apr. 1985, p. 208.
[32] R. S. Sandhu, "Role-based access control," in *Advances in Computers*, vol. 46. Amsterdam, The Netherlands: Elsevier, 1998, pp. 237–286.
[33] S. Nakamura, L. Ogiela, T. Enokido, and M. Takizawa, "An information flow control model in a topic-based publish/subscribe system," *J. High Speed Netw.*, vol. 24, no. 3, pp. 243–257, Jun. 2018.
[34] G. Stergiopoulos, P. Kotzanikolaou, M. Theocharidou, and D. Gritzalis, "Risk mitigation strategies for critical infrastructures based on graph centrality analysis," *Int. J. Crit. Infrastruct. Protection*, vol. 10, pp. 34–44, Sep. 2015.
[35] N. Sahani, R. Zhu, J.-H. Cho, and C.-C. Liu, "Machine learning-based intrusion detection for smart grid computing: A survey," *ACM Trans. Cyber-Phys. Syst.*, vol. 7, no. 2, pp. 1–31, Apr. 2023, doi: 10.1145/3578366.
[36] R. Taormina, "Battle of the attack detection algorithms: Disclosing cyber attacks on water distribution networks," *J. Water Resour. Planning Manage.*, vol. 144, no. 8, Aug. 2018, Art. no. 04018048.
[37] J. Goh, S. Adepu, K. N. Junejo, and A. P. Mathur, "A dataset to support research in the design of secure water treatment systems," in *Proc. Int. Conf. Critical Inf. Infrastructures Secur.*, Paris, France, Jan. 2017, pp. 88–99, doi: 10.1007/978-3-319-71368-7_8.
[38] C. M. Ahmed, V. R. Palleti, and A. P. Mathur, "WADI: A water distribution testbed for research in the design of secure cyber physical systems," in *Proc. 3rd Int. Workshop Cyber-Phys. Syst. Smart Water Netw.*, Pittsburgh, PA, USA, Apr. 2017, pp. 25–28, doi: 10.1145/3055366.3055375.
[39] G. Stergiopoulos, P. Kotzanikolaou, M. Theocharidou, G. Lykou, and D. Gritzalis, "Time-based critical infrastructure dependency analysis for large-scale and cross-sectoral failures," *Int. J. Crit. Infrastruct. Protection*, vol. 12, pp. 46–60, Mar. 2016.
[40] S. R. Center, "Smartgrid resource center & use case repository," in *Use Case Repository*. Palo Alto, CA, USA: EPRI, 2010. [Online]. Available: https://smartgrid.epri.com/repository/repository.aspx
[41] A. Anagnostopoulou, D. Gritzalis, I. Mavridis, and P. Kantas, "Smart environments: Information flow control in smart grids," in *Secur. Privacy Smart Environments*. Cham, Switzerland: Springer, 2024.
[42] K. Adamos, G. Stergiopoulos, M. Karamousadakis, and D. Gritzalis, "Enhancing attack resilience of cyber-physical systems through state dependency graph models," *Int. J. Inf. Secur.*, vol. 23, no. 1, pp. 187–198, Feb. 2024.
[43] A. Dwivedi, X. Yu, and P. Sokolowski, "Analyzing power network vulnerability with maximum flow based centrality approach," in *Proc. 8th IEEE Int. Conf. Ind. Informat.*, Jul. 2010, pp. 336–341.
[44] A. Bavelas, "Communication patterns in task-oriented groups," *J. Acoust. Soc. Amer.*, vol. 22, no. 6, pp. 725–730, Nov. 1950.
[45] H.-H. Chen and U. Dietrich, "Normalized closeness centrality of urban networks: Impact of the location of the catchment area and evaluation based on an idealized network," *Appl. Netw. Sci.*, vol. 8, no. 1, p. 60, Sep. 2023, doi: 10.1007/s41109-023-00585-0.
[46] L. C. Freeman, "Centrality in social networks conceptual clarification," in *Social Network: Critical Concepts in Sociology. Londres: Routledge*, vol. 1. Evanston, IL, USA: Routledge, Jan. 1978, pp. 215–239.
[47] P. Fraga-Lamas, T. Fernández-Caramés, M. Suárez-Albela, L. Castedo, and M. González-López, "A review on Internet of Things for defense and public safety," *Sensors*, vol. 16, no. 10, p. 1644, Oct. 2016.
[48] Y. Zhang, B. Wang, C. Wu, X. Wei, Z. Wang, and G. Yin, "Attack graph-based quantitative assessment for industrial control system security," in *Proc. Chin. Autom. Congr. (CAC)*, Nov. 2020, pp. 1748–1753.
[49] S. Boulares, K. Adi, and L. Logrippo, "Information flow-based security levels assessment for access control systems," in *Proc. 6th Int. Conf. E-Technol.*, Jan. 2015, pp. 105–121, doi: 10.1007/978-3-319-17957-5_7.
[50] Y. Zhang, Z. Wang, Y. Wang, K. Lin, T. Li, H. Liu, C. Li, and B. Wang, "A risk assessment model for similar attack scenarios in industrial control system," *J. Supercomput.*, vol. 79, no. 14, pp. 15955–15979, Sep. 2023, doi: 10.1007/s11227-023-05269-1.
[51] M. Fujimoto, W. Matsuda, T. Mitsunaga, and Y. Hashimoto, "Efficient industrial control systems risk assessment using the attack path to the critical device," in *Proc. 3rd Int. Conf. Manage. Sci. Ind. Eng.*, Osaka, Japan, Apr. 2021, pp. 104–110, doi: 10.1145/3460824.3460859.
[52] P. Buczkowski, P. Malacaria, C. Hankin, and A. Fielder, "Optimal security hardening over a probabilistic attack graph: A case study of an industrial control system using the CySecTool tool," 2022, *arXiv:2204.11707*.
[53] S. Maragkou and A. Jantsch, "Information flow tracking methods for protecting cyber-physical systems against hardware trojans—A survey," 2023, *arXiv:2301.02620*.

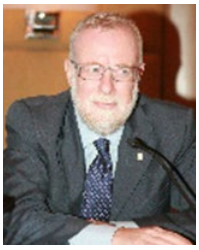


ARGIRO ANAGNOSTOPOULOU received the B.Sc. degree in informatics and the M.Sc. degree in information systems from Athens University of Economics and Business (AUEB), Greece, where she is currently pursuing the Ph.D. degree with the Department of Informatics. She is a Researcher with the INFOSEC Research Group, AUEB. She has authored and co-authored several research papers in international journals and conferences. She has been actively involved in various research projects focusing on cybersecurity and critical infrastructure protection. Her current research interests include the cybersecurity of information and industrial control systems, the protection of critical infrastructures, and the enhancement of access control through the information flow control. Finally, she served as a reviewer for numerous scientific journals and conferences.

IOANNIS MAVRIDIS received the Diploma degree in computer engineering and informatics from the University of Patras, Greece, and the Ph.D. degree in information systems security from the Aristotle University of Thessaloniki, Greece. He is currently a Professor of information security with the Department of Applied Informatics, University of Macedonia (UoM), Greece. He is also the Director of the Multimedia, Security and Networking Laboratory (MSN Lab). His research interests include AI-based attack detection, cybersecurity education, risk management, access control, cyber threat intelligence, digital forensics, and security economics. He serves as an Area Editor for *Array* (Elsevier).

MICHAEL ATHANASOPOULOS received the Diploma degree in electrical and computer engineering from the National Technical University of Athens (NTUA), and the M.Sc. degree in information systems from the Athens University of Economics and Business (AUEB), Greece. He is currently a Cybersecurity Analyst focused on the Blue Team, actively fortifying corporate defenses and bolstering resilience against cyber threats that is further assisted from his strong background in IT which enhances his ability to comprehend complex systems. His role encompasses monitoring and analyzing security alerts, investigating potential security incidents, identifying and responding to threats, and implementing measures to mitigate risks within organizations information technology infrastructures. His current research interests include the cybersecurity of information and industrial control systems, the protection of critical infrastructures, and the enhancement of access control through the information flow control.

ALEXIOS MYLONAS received the B.Sc. degree (Hons) in computer science from the Athens University of Economics and Business, the M.Sc. degree in information security from the Royal Holloway, University of London, and the Ph.D. degree in information and communication security from the Athens University of Economics and Business. He is currently with the University of Hertfordshire, where he leads the Cybersecurity and Computing Systems Research Group. He has published more than 40 articles in esteemed scientific venues. His research interests include the IoT security, incident response and web security, and fraud detection.



DIMITRIS GRITZALIS received the B.Sc. degree in mathematics from the University of Patras, Greece, the M.Sc. degree in computer science from the City University of New York, USA, and the Ph.D. degree in information systems security from the University of the Aegean, Greece. He is currently a Professor of cybersecurity with the Department of Informatics, Athens University of Economics and Business (AUEB), Greece, where he is also the Director of the M.Sc. Program in Information Systems Security. His current research interests include cybersecurity, critical infrastructure protection, risk assessment, malware, access control, and security education. He served as an Associate Rector for Research (AUEB), the President for Greek Computer Society (GCS), and the Associate Data Protection Commissioner for Greece. He is the Academic Editor of *Computers and Security* (Elsevier) and the Scientific Editor of the *International Journal of Critical Infrastructure Protection* (Elsevier).